به نام خدا

کنترل سیستم های بلادرنگ

دکتر سهیل سید حسینی

دانشجو:حسین عبدلی

مقدمه:

واژه سیستم بی‌درنگ در اکثر موارد متأسفانه اشتباه استفاده می‌شود. این معنی در استاندارد DIN آلمان با شماره 44300 صحنه دوم خط پنجم اینگونه تعریف می‌شود.

عمل بی‌درنگ **عبارتست از عملکرد یک سیستم کامپیوتری که در آن برنامه‌های لازم برای پردازش داده‌های وارده از خارج سیستم همواره آماده باشد بگونه‌ای که اجراء شدن آن آسان بوده و نتایج آن در فواصل زمانی بطوری که از قبل تعیین شده باشند را قابل حصول می‌کند،این عمل بسته به نوع زمان و ورود داده‌ها می‌تواند بصورت تصادفی و از قبل تعیین شده باشد.**

بنابراین وظیفه کامپیوترهای دیجیتال در این وضعیت این است که برنامه‌های ما را اجراء کند و با رویدادهایی که در خارج اتفاق می‌افتد و طبق برنامه می‌باشد، عمل نماید. سیستم‌های بلادرنگ همیشه بعنوان بخشی از یک محیط بزرگتر که آن را در برگرفته است در نظر گرفته می‌شود از این رو در کتاب‌های آمریکا (دانشگاه یوتا) آن را بعنوان سیستم نهفته نیز می‌نامند (embedded system) بطوری که پدیده عملکرد بی‌درنگ را می‌توان با مثال‌هایی از زندگی روزمره نشان داد:

راننده‌ای که در اتوبان خودروی خود را می‌داند باید موظف پیاده‌ها، سایر خودروها، حیوانات و چراغ راهنمایی باشد و همواره آماده باشد تا با یک سرعت قابل اطمینان از حادثه جلوگیری کرده و عکس‌العمل نشان دهد. اساساً راننده‌هایی که بطور تصادفی سیگنال‌هایی را از محیط اطراف دریافت کرده، پردازش اطلاعات و انجام عکس‌العمل آنها منجر به تصادف خواهد شد.

جدول ۱- محیط‌های بی‌درنگ را مورد دسته‌بندی قرار داده و با استفاده از مثال‌های ساده آنها را دسته‌بندی می‌کند. استاندارد DIN آلمان شرایط بی‌درنگ را با دو حالت سخت یا

نرم (hard or soft real time) دسته‌بندی می‌کند. ما سعی می‌کنیم ساختار و جنبه‌های تاریخی سیستم‌های بی‌درنگ را بررسی کرده، عناصر اصلی را شناخته و یک زبان برنامه‌نویسی ساده برای آنها داشته باشیم. در درس کنترل کننده‌های برنامه پذیر با کارکرد دقیق آنها بیشتر آشنا می‌شویم.

جدول I

کنترل کننده‌های بلا درنگ

نرم
سخت

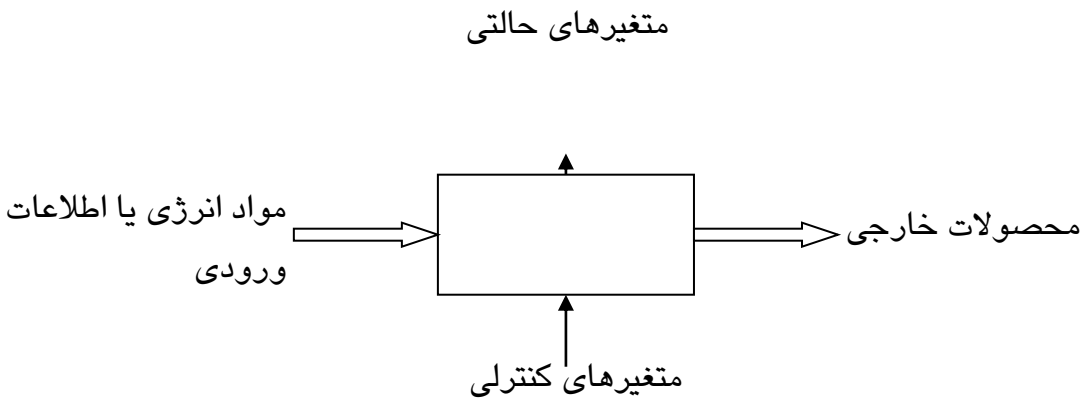
کنترل کننده‌های صنعتی:

در بحث بی‌درنگ دو مفهوم اساسی فرآیند و فرآیند صنعتی تعریف می‌شود که طبق استاندارد DIN 6620 آلمان به این صورت تعریف می‌شود.

فرآیند: عبارتست از کلیه فعالیت‌های یک سیستم که بر یکدیگر اثر گذاشته و با آن ماده، انرژی یا اطلاعات را تبدیل، منتقل یا ذخیره می‌کند.

فرآیند صنعتی: فرآیندی است که کمیت‌های فیزیکی آن را بتوان با وسایل فنی بدست آورد و تحت تأثیر قرار داد. فرآیند صنعتی را می‌توان به فرآیندهای تولیدی، توزیعی، ذخیره‌ای طبقه‌بندی کرد. به شکل صفحه بعد نگاه کنید با تعاریفی که انجام شد فرآیند در داخل یک محفظه نشان داده شده و این فرآیند می‌تواند مجموعه‌ای از لوله‌ها، دریچه‌ها و ستون‌های تبخیر را در یک پالایشگاه شامل می‌شود. در این جریان قطعات و

موادی وارد می‌شوند تا تبدیل، انتقال، یا ذخیره شوند. این جریان در شکل بصورت یک پیکان با دو خط افقی نشان داده شده است.



برای نگهداری فرآیند در یک حالت دلخواه باید آن را مطابق با قوانین معینی کنترل کرد این قوانین بوسیله سیستم کنترلی ایجاد می‌شود که ممکن است اپراتوری (دستی)، خودکار یا ترکیبی از این دو باشد واژه اتوماتیک مترادفی با بکارگیری کامپیوتر است. دلیل استفاده از سیستم‌های کنترل کامپیوتری این است که بسیار قابل اعتماد، دقیق و ارزان هستند. سیستم‌های کنترل باید به نحوی به سیستم‌های فرآیند متصل شوند تا مقادیر متغیرهای حالت را خوانده و آنها را تنظیم کنند بسته به نوع اتصال سیستم‌های کامپیوتری را می‌توان به سیستم‌های (offline) ناپیوسته، پیوسته با حلقه باز (online with open loop) پیوسته با حلقه بسته طبقه‌بندی کرد.

جدول ۲-۱- طبق استاندارد DIN انواع پیوند بین فرآیند و سیستم‌های کنترل کامپیوتری را نشان می‌دهد.

ارتباط کامپیوتر با فرآیند	ارتباط فرآیند با کامپیوتر	نوع
متصدی انسانی	متصدی انسانی	ناپیوسته
تزویج مستقیم	متصدی انسانی	پیوسته، حلقه باز
متصدی انسانی	تزویج مستقیم	پیوسته و حلقه باز
تزویج مستقیم	تزویج مستقیم	پیوسته و حلقه بسته

به جدول ۲-۱ نگاه کنید. سیستم‌هایی که ورودی دستی دارند در دو ردیف اول جدول جزء سیستم‌های بی‌درنگ محسوب نمی‌شوند زیرا مقیاس زمانی برای اندازه‌گیری‌های دستی و عملیات ورودی از نظر استانداردهای کامپیوتری آهسته هستند. سیستم‌های ناپیوسته را اغلب سیستم‌های کنترلی پیوسته با ورودی دستی به ندرت استفاده می‌شود.

سیستم‌هایی که در آن تزویج مستقیم (Direct Coupling) بین فرآیند و کامپیوتر وجود دارد در دو ردیف آخر جدول بطور مستقیم جزء مباحث کنترل بلادرنگ هستند و بخش عمده‌ای در چنین سیستم‌هایی بکار می‌روند عبارتند از:

۱- اکتساب داده‌ها و نظارت آنها (مانیتورینگ) و استفاده از مدار باز

۲- کنترل فرآیند و تولید خودکار با استفاده از مدارات بسته کنترل

سیگنال‌های تولید شده بوسیله حسگرها و عملگرها متصل به تأسیسات فرآیند بطور کلی با استانداردهای برقی و منطقی مطابقت ندارند و یک رابطه کنترلی، کامپیوتر بسته به نوع فرآیند و تجهیزات مربوطه می‌تواند به یکی از اشکال زیر باشند:

۱- پیوسته (آنالوگ)

۲- دو حالته (دورویی)

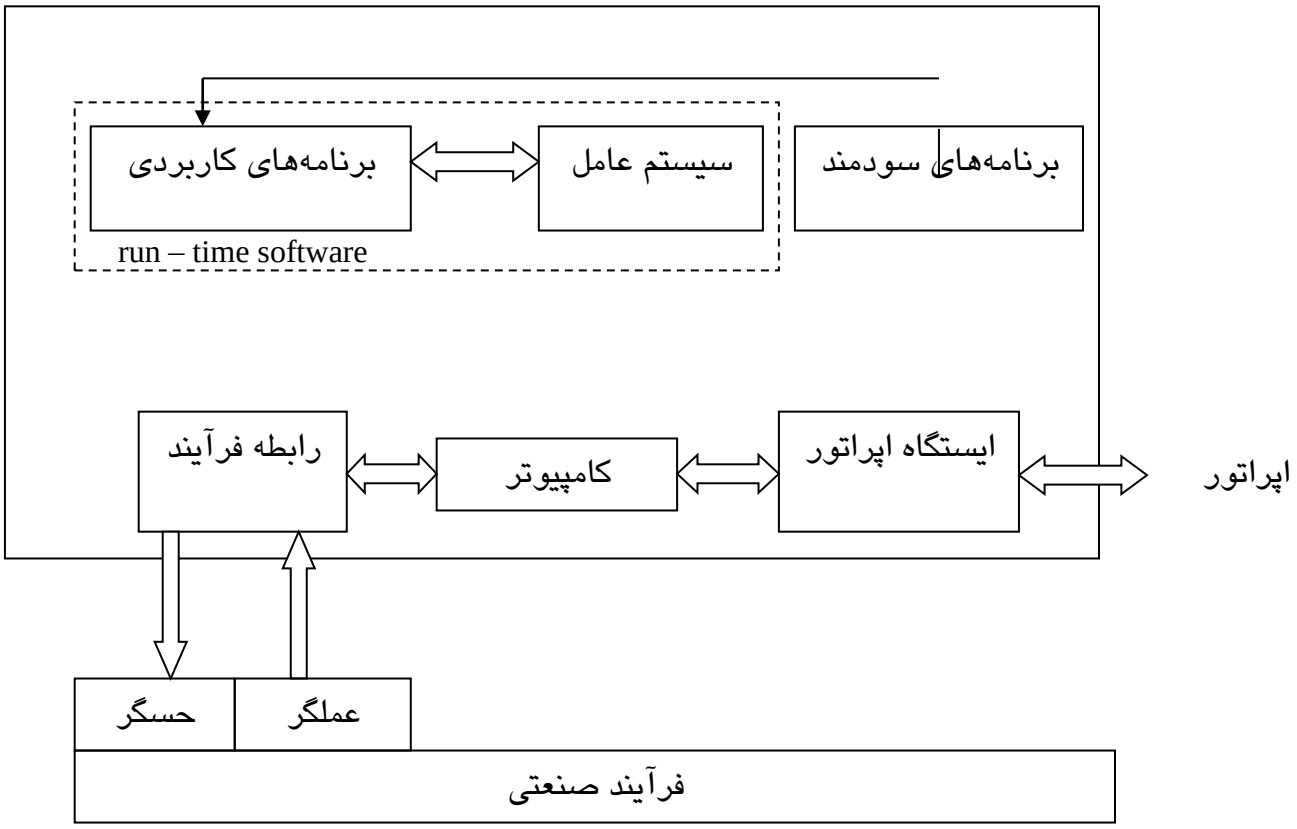
۳- پالسی (طپشی)

پیوسته: سیگنال‌هایی که بطور پیوسته پارامترهای آن با زمان تغییر می‌کند و این تغییرات پیوسته، مقادیر متغیرهای فرآیند را منعکس می‌کند.

دو حالته: سیگنال‌هایی دودویی، مقادیر جاری متغیرهای دودویی را منعکس کرده که اغلب با ولتاژ کم کار می‌کنند (مثل کنتاکتورها)

پالسی: سیگنال‌هایی را بصورت پویا تولید می‌کند که تغییرات آنها منطبق بر تغییرات فرآیند است این سیگنال‌ها می‌توانند کمتر از چند هزارم ولت یا بیشتر از چندین ولت باشند.

عناصر سخت افزاری یک سیستم بی‌درنگ در شکل ۲ نشان داده شده است.



جدول مراحل اصلی در تکنولوژی سیستم‌های بی‌درنگ

روش	نرم افزار	سخت افزار
روش‌های دلخواه	بدون سیستم عامل	کامپیوترهای متمرکز بزرگ/ کابل کشی ستاره‌ای شکل پایانه‌های ابتدایی/ رابط فرآیند ابتدایی
برنامه سازی ساخت یافته	برنامه‌های اجرائی چند تکلیفی زبان‌های سطح بالا	مینی کامپیوترها - سیستم‌های چند کامپیوتری – ساختارهای گذرگاهی پایانه‌های گرافیکی
طراحی ساخت یافته ابزار CASE تحلیل ساخت یافته	رابط خوشکار به برنامه‌های سیستمی بسته برنامه‌های کاربردی	ریز کامپیوترها – شبکه‌های توزیع یافته – پایانه‌های هوشمند رابط فرآیند هوشمند
ابزار ساخت و تولید نرم افزار برای چرخه کامل زندگی	سیستم‌های عامل هوشمند زبان‌های بی‌درنگ توزیع یافته	1990

مفاهیم اساسی در مشخصات سیستم‌های بی‌درنگ:

پردازش بی‌درنگ بعلت داشتن بعد زمان با سایر انواع داده پردازی‌ها متفاوت است این تفاوت با دو نیاز اساسی کاربر که باید با سیستم‌های بی‌درنگ برآورده شوند بیان می‌شود که:

۱- به موقع بودن

۲- هم زمانی

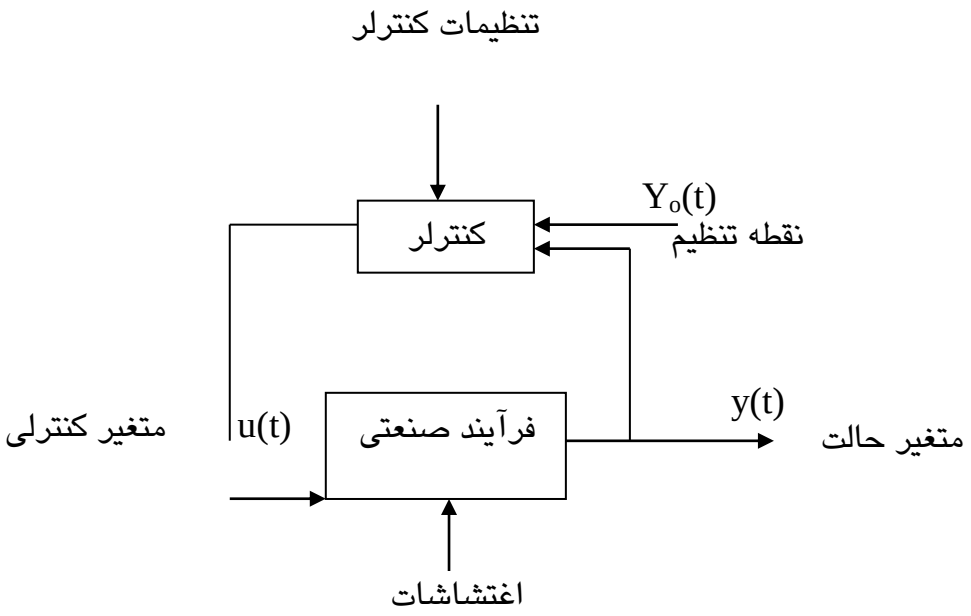
این دو نیاز هستند.

سیستم‌های مختلفی وجود دارند که به ورودی از چندین منبع نیاز دارند و ممکن است در مورد زمان نسبت به هم همپوشانی داشته باشند ولی نیاز واقعی برای همزمانی معمولاً مستلزم پردازش همبسته می‌باشد که در این سیستم بیش از یک ورودی در یک فاصله زمانی وجود دارد بعنوان مثال یک هواپیما ممکن است مشخصات سرعت و موقعیت را بصورت همبسته و حرارت و رطوبت داخل کابین را بصورت هم روند کنترل نماید در یک سیستمی ممکن است به محض دریافت یک تقاضا از یک فرآیند که بصورت خارجی صادر شده است باید اکتساب و ارزیابی اطلاعات را بصورت عکس‌العمل‌های مناسبی برای سیستم انجام داد. در سیستم‌های بی‌درنگی که سخت افزار دارند باید حتی در بدترین شرایط کاری این نیازها فراهم شود. این دو نیاز مهم قابلیت وابستگی است.

علیرغم بهترین برنامه‌ریزی در یک سیستم ممکن است یک وضعیت emergency به صورت اضافه بار یا overload در یک گره بوجود آید. برای رفع چنین مشکلی در موارد خاص از روش‌های تقسیم بار استفاده می‌کنند که در یک سیستم گسترده تکالیف از یک گره به گره دیگر منتقل می‌شود. بنابر تعریف این سیستم‌ها دارای وابستگی بالایی هستند زیرا آمادگی کامپیوترها در این حالت بیشتر است.

سیستم‌های با زمان گسسته و زمان پیوسته:

سرعت پاسخ کامپیوتر را می‌توان بر حسب وابستگی بین ما و رویدادهایی که واقع شده‌اند بیان کرد بعنوان مثال خط بطری پرکن کارخانه‌ای را در نظر بگیرید که هر وقت بطری پری در خط قرار گیرد دیگر در آن مایعی ریخته نمی‌شود. توجه به زمان محتویات دارای تلرانس ۱۵ $\pm$ میلی‌لیتر باشد زمان پر کردن آن از 50 میلی ثانیه نمی‌تواند بزرگتر شود. شکل زیر فرآیند زمان پیوسته را نشان می‌دهد.



در سیستم‌های زمان پیوسته وضعیت کاملاً متفاوت است که متغیرهای حالت و کنترلی بصورت توابعی پیوسته‌ای از زمان هستند y(t) و u(t) آن دو را در شکل نشان می‌دهد. یک کنترلر ایده‌آل کنترل کننده‌ای است که توابع y(t) را خوانده و تابع u(t) را تولید کند.

در سیستم‌های فرآیند زمان پیوسته کامپیوترها سه تکرار متفاوت دارند که در هر فاصله زمانی بصورت Δt انجام می‌دهند.

۱- ورودی ۲- ارزیابی ۳- خروجی

۱- ورودی: مقدار جاری $y(t)$ در لحظه t و متغیر حالتی از فرآیند خوانده می‌شود که بتواند بیشترین مقدار ورودی را به سیستم داشته باشد.

۲- ارزیابی: مقدار جدید متغیر کنترلی را محاسبه می‌کند.

۳- خروجی: در لحظه زمانی $t+T$ (لحظه‌ای که آغازش از مرکز نیست) مقدار محاسبه شده u در $t+T$ یک مقدار کنترلی بوده که بصورت داده به کنترلر ارسال شده و از آنجا خارج شود. مقدار خروجی $u(t+T)$ در فاصله زمانی Δt زمانی که نمونه بعدی آن محاسبه شده، ثابت نگه داشته می‌شود.

روش‌های مهندسی در طراحی سیستم‌های بی‌درنگ:

در ساخت سیستم‌های مستدل بی‌درنگ طبق استاندارد DIN موارد زیر در نظر گرفته می‌شود که:

۱- طراحی نرم افزار و سیستم بصورت یکپارچه بوده و از ابتدا با شرح تمام رفتارهای احتمالی را در نظر می‌گیریم.

۲- ارزیابی کارایی با مبنای درک روشن از حداکثر کارهای یک سیستم حتی قبل از طراحی و پیاده سازی سیستم

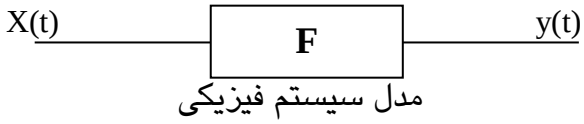
۳- از روش‌هایی که بتوان شرح مختصاتی به صورت ریز دانه از آزمون پذیری و پیاده سازی مشخصات سیستم استفاده کرد به نحوی که یک استاندارد مطمئن کارکرد آن را تضمین کند.

۴- از روش‌ها و ابزارهای طراحی و پیاده‌سازی که بتواند سیستم‌های توزیع یافته‌ای را که قابلیت تحمل خطا را به صورت مداوم داشته باشند استفاده کنند.

۵- از معماری سیستم و سخت افزار توزیع یافته‌ای که دارای سیستم عامل قابل اطمینان بوده و محدودیت ضایع نداشته باشد استفاده کنیم.

کنترل دیجیتال در فرآیند پیوسته:

سیستم‌های پیوسته را غالباً آنالوگ می‌گویند که امروزه با استفاده از کامپیوترهای که حالت سیستم را ارزیابی کرده و سیگنال‌های کنترلی دقیق را تولید می‌کنند عمل کنترل به سیستم‌های دیجیتالی واگذار شده است. در سیستم‌های دیجیتالی می‌توان از آنها به عنوان اطلاعات زمینه‌ای جهت سایر موضوعات از قبیل نمونه برداری و کوانتیزه کرده استفاده کرده، یک مدل فیزیکی در یک سیستم خطی با یک جعبه سیاه F و سیگنال ورودی X(t) و خروجی y(t) وجود دارد. می‌دانید سیستم در صورتی خطی است که شرایط زیر برای دو سیگنال ورودی یعنی $X_1(t)$ و $X_2(t)$ ثابت بوده و مقادیر F(x(t)) اینگونه باشد.



مدل سیستم فیزیکی

$$\begin{cases} F\left(x_1(t)+x_2(t)\right)=Fx_1(t)+Fx_2(t) \\ \lambda Fx_1(t)=F\left(\lambda x_2(t)\right) \end{cases}$$

سیستم‌های تغییر ناپذیر با زمان را در استاندارد DIN خطی می‌گویند و به دو طبقه پیوسته زمانی و گسسته تقسیم‌بندی می‌شوند. در سیستم‌های پیوسته زمانی تابع با X(t) و X(t) در ارتباط است که آن را می‌توان با معادلات دیفرانسیل به راحتی حل کرد. در مورد سیستم‌های گسسته زمانی روابطی وجود دارد که تجزیه آن به صورت زیر است:

$$X[K]=\sum_{h=-\infty}^{\infty}\times[n]\delta[k-n]$$ **رابطه I**

$$y[K]=\sum_{h=-\infty}^{\infty}\times[n]h[k-n]$$ **رابطه II**

$$H[s]=\frac{y(s)}{x(s)}$$ **رابطه III**

$$H[t]=\frac{y(t)}{x(t)}$$ **رابطه IV**

$$H[s]=\frac{Q_o(s)}{Q_i(s)}=\frac{۱}{s+SRC}$$ **رابطه V**

به رابطه I نگاه کنید با تجزیه دقیق این رابطه می‌توان سری اعداد کراندار را طبق فرمول شماره II تجزیه کرده و خروجی ستیستم را با استفاده از قوانین کانولوشن به صورت یک ضرب بیان کرد. در رابطه شماره III که پردازش سیگنال‌های دیجیتالی مستقیماً از این الگوریتم استفاده می‌شود در الگوریتم‌های سیستم کنترلی استفاده کرده و با تشریح یک سیستم به رابطه شماره III که یک ترانسفرفانکشن (Function) انتقالی بوده و برای سیستم‌های پیوسته و خطی استفاده می‌شود بکار برد.

به فرمول IV نگاه کنید. این فرمول که زیر مجموعه III می‌باشد X(s) و y(s) به ترتیب تبدیل لاپلاس X(t) و y(t) می‌باشند که برای سیستم‌های گسسته زمانی در تبدیل t و سیستم‌های دیجیتالی کاربرد دارد. در نتیجه رابطه چهارم یک تابع انتقالی برای نمایش

سیستم در حوزه فرکانس بالا می‌باشد بدست می‌آید. در معادلات دیفرانسیل پاسخ ضربه نمایش سیستم در حوزه زمان است.

حوزه‌های فرکانسی مزیت اساسی نسبت به حوزه‌های زمانی دارند که به جای مشتق‌گیری یا انتگرال‌گیری پاسخ را می‌توان بصورت عملیاتی بدست آورد.

به رابطه V نگاه کنید در این رابطه یک رابطه مستقیم بین پاسخ ضربه H(s) و تبدیل لاپلاس H(t) وجود دارد. H(t) تبدیل به H(Kt) می‌شود و با استفاده از تبدیلات عکس می‌توان با گرفتن تبدیل لاپلاس معادله مخصوص به آن را بدست آورد توجه به این نکته لازم است از این سیستم در یک صافی بالا گذر که فرکانس‌های پائین را حذف کرده و فرکانس‌های بالا را عبور می‌دهند استفاده می‌شود پس اینگونه نتیجه بگیریم که تبدیل لاپلاس در حل کردن توابع کنترلی و تبدیل آن بسیار مهم بوده و تغییرات در حوزه زمان را کنترل می‌کند.

طراحی و تحلیل در سیستم کنترل:

وظیفه اصلی یک سیستم از نوع کنترل در محیط صنعتی آن است که فرآیند را در رسیدن به یک محصول با کیفیت بالا کمک کند.

در تحلیل و طراحی سیستم کنترل عناوین ریشه‌ای زیر را باید در نظر گرفت:

۱- کنترل پذیری (کنترل ability): فرآیندی کنترل پذیر است که فقط و فقط با تغییر دادن ورودی‌ها در یک زمان به صورت محدود بتوان در هر یک از حالت‌های داخلی دسترسی پیدا کرد.

۲- مشاهده پذیری (absorb ability): سیستم کنترل باید بتواند با استفاده از مقادیر ورودی سیستم را کنترل کرده و کارکرد آن قابل مشاهده باشد.

۳- پایداری: کل سیستم باید طوری پایدار باشد تا یک سری ورودی محدود منجر به رشد خروجی‌های نامحدود این سیستم به صورت کلی پایدار است.

۴- رفتار سیستم: بطور کلی پاسخ کل سیستم برای ورودی‌های خارجی آن مشخص می‌شود رفتار سیستم باید طوری باشد که مشخصات طراحی را برآورده کند.

نکته: از چهار مورد بالا سه مورد اول نیازهای اصلی هر سیستم عمل هستند که باید برآورده شود.

حال به تعریف و تحلیل سیستم می‌پردازیم:

در تحلیل سیستم‌ها، باید نیازهای مشخصی را که سیستم برای آن طراحی شده در نظر گرفته و بهترین کارکرد سیستم را بعنوان کارکرد بهینه در ماکزیمم ضریب بهره تحلیل نمود.

نمودار زیر که یک نمودار بلوکی از یک سیستم کنترل بسته می‌باشد به صورت کلی مورد تحلیل قرار گرفته و به کمک آن کلیه توابع کنترل کننده‌های تناوبی، انتگرالی و مشتق‌گیر را می‌توان بررسی کرد. با توجه به شکل می‌بینید که سیگنال‌ها به صورت:

R(s) ورودی مرجع

B(s) سیگنال بازخور (سیگنال فیدبک)

E(s) سیگنال خطا

M(s) سیگنال تحریک

C(s) سیگنال خروجی

می‌باشد با توجه به بلوک‌ها بلوک Gc(s) تابع انتقال کننده Gp(s) تابع انتقال فرآیند و H(s) تابع انتقالی فیدبک می‌باشد.

اگر بخواهیم همه سیگنال‌ها و بلوک‌ها را در حوزه فرکانس نمایش دهیم اینگونه می‌توان با استفاده از معادلات آنها را به یکدیگر مرتبط کرد که:

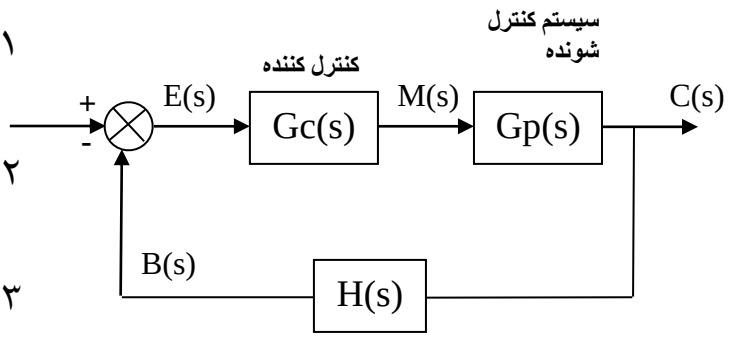
$$E(s) = R(s) - B(s)$$
$$C(s) = G(s) \cdot E(s)$$
$$B(s) = H(s) \cdot C(s)$$

می‌باشد. در فرمول‌های بالا G(s) برابر با ضرب Gc(s) و Gp(s) می‌باشد که تابع انتقال در حلقه باز است. از این رو می‌توان با استفاده از مفاهیم بالا و نمودار بلوکی سیستم کنترل به معادله‌های زیر دست یافت در زیر معادله‌ها و نمودار بلوکی را می‌بینید:

$$\frac{C(s)}{G(s)} = R(s) - H(s)\,C(s) \qquad ۱$$

$$C(s)\left(\frac{۱}{G(s)} + H(s)\right) = R(s) \qquad ۲$$

$$\frac{C(s)}{R(s)} = \frac{G(s)}{۱ + G(s)\,H(s)} \qquad ۳$$



نمودار بلوکی یک سیستم کنترل بسته

به معادله ۳ نگاه کنید، این معادله تابع انتقالی حلقه بسته را بصورت کلی نشان می‌دهد و نشان می‌دهد که رفتار یک حلقه بسته بویژه با پاسخ سیستم C(s) به سیگنال ورودی R(s) مشخص می‌شود و با تغییر دادن Gc(s) و یا H(s) می‌توان تابع انتقال کلی را تغیر داده تا اینگونه بتوانیم رفتار سیستم را تغییر دهیم.

هدف از طراحی چنین سیستم‌هایی تعریف توابع Gc(s) و H(s) می‌باشد بطوری که تضمین کند سیستم رفتار مطلوب دارد. رفتار مطلوب یک سیستم به نظر طراح در این سه مشخصه خلاقه می‌شود.

۱- مشخصه پاسخ سیستم: این کار را می‌توان در حوزه زمان یا حوزه فرکانس انجام داد. در حوزه زمان غالباً مشخص می‌شود که پاسخ سیستم به ورودی پله چیست و این پاسخ بر حسب حداکثر مقدار (over shoot) (حداکثر جهش) زمان صعود rais time زمان تأخیر Delay time زمان نشست (setting)، نسبت میرایی Damping و فرکانس طبیعی مشخص می‌شود.

در حوزه فرکانس موقعیت قطب‌ها در صفحه‌ها پهنای باند یا شکل تابع آن می‌تواند مدنظر باشد.

۲- مشخصه پاسخ حالت پایدار سیستم: یعنی حالتی از سیستم که پس از مدتی که از آن سپری شد پاسخ حالت پایدار غالباً برای ورودی‌های متفاوتی مثل پله یا شیب (ramp) به صورت $r(t) = c(t) \quad t \geq ۰$ در tهای بزرگتر یا مساوی صفر باشند معین می‌شود.

در کتاب‌های آمریکایی ارزیابی پاسخ حالت پایدار برای سیستم‌های servo بسیار مهم است یعنی برای سیستم‌هایی که خروجی آنها ورودی‌های معینی را دنبال کند مثل سیستم‌های ردیاب خورشیدی بسیار مهم است.

۳- مشخصه حساسیت سیستم نسبت به تغییرات:

طراحی سیستم‌هایی که بعد از تغییر کوچکی پارامترهای آن دچار مشکل شده و سیستم بطور محسوس تغییر می‌کند بسیار بی‌فایده بوده و حساسیت سیستم‌ها به صورت کاربردی باید به انواع تغییرا بسیار کم باشد.

در صنعت با استفاده از سیستم حلقه باز با استفاده از کنترل کننده‌های تناسبی، انتگرالی و مشتق‌گیر بسیار استفاده می‌شود. در جدول زیر مشخصات این سه نوع کنترل کننده در حوزه زمان یا حوزه فرکانس با تابع انتقالی کنترل کننده مشخص شده است. از ترکیب این سه نوع کنترل کننده PID حاصل می‌شود که ئر عمل کنترل از بقیه کامل‌تر است.

نوع کنترل	حوزه زمان	حوزه فرکانس
تناسبی	$m(t) = kpe(t)$	$Gc(s) = kp$
انتگرالی	$m(t) = ki\int_{-\infty}^{t} e(t)dt$	$Gc(s) = \frac{ki}{s}$
مشتقی	$m(t) = kd\frac{de(t)}{dt}$	$Gc(s) = kp + kds + \frac{ki}{s}$

کنترلر PID

$$m(t) = kpe(t) + ki\int_{-\infty}^{t} e(t)d(t) + kd\frac{de(t)}{dt}$$

حال به بررسی نکاتی اجمالی در سه کنترلر بالا می‌پردازیم:

۱- کنترلر تناسبی: در این کنترلر سیگنال تحریک متناسب با سیگنال خطاست و ضریب بهره Kp مطابق با نیاز انتخاب می‌شود. برای آنکه خطای جهشی و خطای حلات پایدار هر دو در محیط و محدود قابل قبولی باشند. غالباً باید یک کنترل کننده سازگار طراحی کنیم حال اگر این امکان نبود باید کنترل کننده را به نوع دیگری از کنترل بسط دهیم.

۲- کنترلرهای انتگرالی: این کنترل کننده به شدت خطای حالت پایدار را کم می‌کند و سیگنال کنترل در این کنترلر غالباً از نوع مربعی بوده و سیگنال تحریک آن متناسب با انتگرال خطاست از نیرو خطا اگر ثابت باشد سیگنال تحریک به صورت خطی افزایش می‌یابد و این کار را از طریق مسیر پس خور (ورودی) اشکال کنترلرهای انتگرالی در نقشه‌ها غالباً بزرگتر از دیگر انواع است و عیبی که این کنترل کننده دارد این است که سیستم به تغییر خطا خیلی آهسته پاسخ می‌دهد. در ضمن برای مقادیر کمی از Ki

(مقادیر بهره) سیستم ممکن است ناپایدار شود و در پاسخ در حوزه فرکانس به صورت خطی عمل نکند.

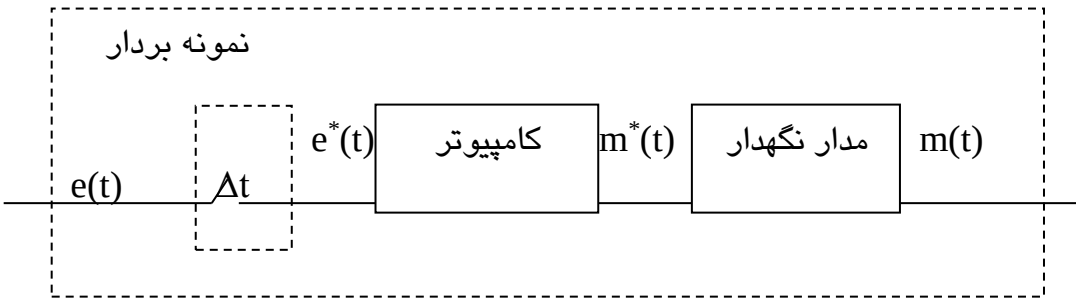
۳- کنترلرهای مشتق‌گیر: می‌دانید که سیگنال تحریک با سرعت تغیر خطا متناسب است و این خود نوعی خود کنترلی ویژه می‌باشد و میرائی سیستم را افزایش می‌دهد.

کنترلر مشتق‌گیر را حتماً باید با ترکیبی دیگر از عناصر کنترلی استفاده کرد چون وقتی خطا ثابت باشد سیگنال تحریک m(t) صفر خواهد ماند سیستم ناپایدار می‌شود.

کوانتیزه کردن دامنه:

به معنی گرد کردن اعداد بدست آمده در مرحله نمونه برداری به نزدیک‌ترین عدد صحیح است از طرف دیگر چون خروجی کامپیوتر را باید به سیگنال بصورت پیوسته تبدیل کرد. این عمل توسط Hold circuitها انجام می‌شود. یک نمونه‌برداری ایده‌آل از سه قسمت تشکیل شده که شامل نمونه بردار، کامپیوتر و یک مدار نگهدار است.

شکل زیر را ببینید. این کامپیوترها در حقیقت می‌توانند با ورودی‌های به شکل مقادیر واقعی کار کنند به شکل نگاه کنید:



اصول سیستم‌های بی‌درنگ:

یک سیستم عامل بی‌درنگ به صورت مجموعه‌ای از عناصر نرم افزاری و اجزاء برنامه‌های کنترلی و کامپیوتری برای اختصاص منابع کامپیوتری فراهم می‌کنیم و بسته به نوع سیستم عامل می‌توان خدماتی اضافی برای شبکه در نظر گرفت، سیستم‌ها عامل بی‌درنگ غالباً توسط شبکه‌های LAN و WAN کنترل می‌شوند که در این محیط رابطه‌ای بین ارسال سیگنال و دستیابی محیط به LANها با انواع توپولوژی‌های شبکه‌ای و خطی و با استفده از مزایای سیستم که سادگی الگوریتم و شرایط کاری سبک می‌باشد یک میدان داده تولید شده که شامل یک Header و یک دنباله بود و آدرس مبداء و آدرس مقصد با استفاده از یک قاب اطلاعاتی و پروتوکل‌های مورد استفاده ارتباط متعدی را جهت استفاده سیستم‌های کنترلی فراهم کرد.

یک سیستم بلادرنگ که برای کارهای کنترلی مورد نظر است به میزان قابل ملاحظه‌ای با سیستم‌های چند منظوره و سیستم‌های عامل شناخته شده متفاوت است.

یک سیستم عامل بلادرنگ باید بتواند رویدادهای داخلی و خارجی سیستم را در محدوده زمانی که از بیش تعیین شده پاسخ دهد، می‌دانید که یک رویداد را به صورت رسمی و غیررسمی به وضعیتی تفسیر کرد که سیستم کامپیوتری با آن محیط به وجود آمده است. تفاوت‌های سیستم‌های عامل بی‌درنگ با مشخصات منابع و مکانیزم مدیریت منابع در ارتباط است.

مدیریت پردازنده:

در یک محیط وظیفه‌ای تکالیف در استفاده از منابع با یکدیگر در رقابت هستند و برای رسیدن به اهداف مشترک در یک کاربرد توزیع شده همکاری دارند. نیازهای یک سیستم

بی‌درنگ با سیستم دیگر متفاوت است و استراتژی زمان‌بندی اجرای تکلیف در آنها متفاوت است.

مدیریت حافظه:

در سیستم‌های کامپیوتری بی‌درنگ از دو جنبه بررسی می‌شود جنبه مهم اول این است که فضای حافظه بین تکالیف مختلف که بوسیله سیستم اجراء می‌شود به اشتراک گذاشته شده و مورد پیچیده‌تر تعویض برنامه‌ها بین داده‌ها، حافظه اصلی و حافظه پشتیبان است. یک حافظه مجازی از پیاده سازی نرم افزار بین سطوح مختلف حافظه پشتیبانی می‌کند.

استراتژی خوب در یک سیستم بلادرنگ اختصاص حافظه در سیستم بوده که این حافظه ایستا در فضای حافظه با بکارگیری تکلیف‌های مختلف با امکاناتی برای جای گشت‌ها و با تقاضای آشکارسازی از هر تکلیف در کنترل می‌تواند باشد. در این سیستم‌ها مجموعه‌های ثابتی از برنامه‌ها در **ROM** ذخیره می‌شود که داده‌هایشان بصورت ایستا در حافظه‌ای با قابلیت دستیابی تصادفی **(RAM)** قرار می‌گیرد. در چنین مواردی به مدیریت حافظه برخورد می‌کنیم که در سیستم‌های **Real time** از اهمیت ثانویه برخوردار هستند.

مدیریت وسیله‌ها (Device manager):

مهمترین وسیله جنبه در یک سیستم کنترل ارتباط دهنده‌ها با فرآیند هستند. چنین وسایلی معمولاً با کاربردهایی سیستم عامل را به وسیله مختلف مرتبط می‌کند. در این سیستم‌ها کامپیوتر میزبان با تهیه خدماتی برای اصلاح، ذخیره و ترجمه برنامه‌های کاربردی یک محیط مناسب برای ایجاد نرم افزار بوجود می‌آورد سیستم هدف که برای

اجرای نرم افزار کاربردی برای محیط در حین اجرا محیطی مناسب بوجود می‌آورد و سیستم عامل باید قابلیت پیکیربندی و معماری سخت افزاری سیستم را داشته باشد.

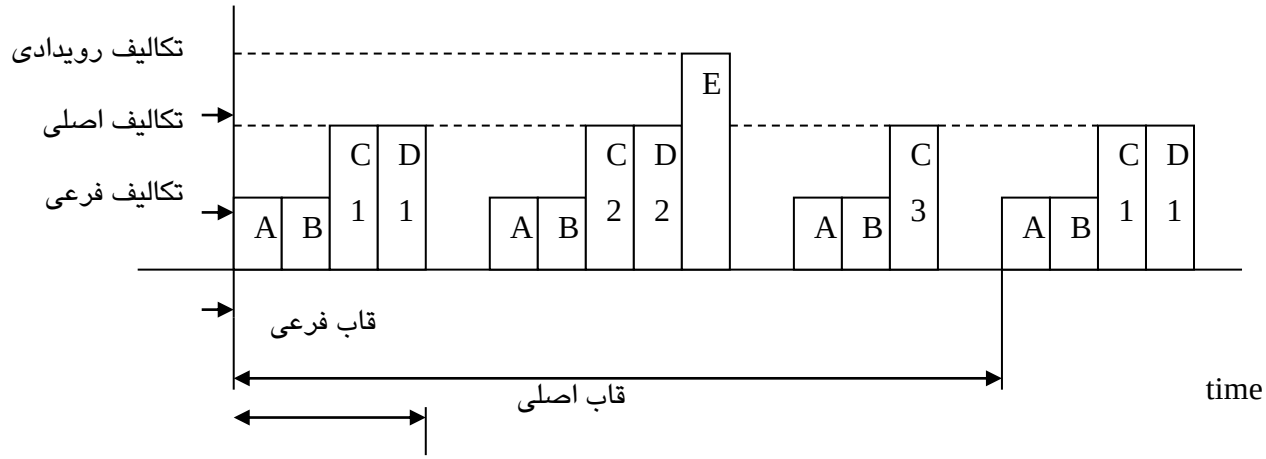
مدیریت فایل‌ها (File manager): مدیریت فایل‌ها و مکانیزم بکار رفته در مدیریت آنها مبتنی بر یک سیستم بی‌درنگ و اراده رویدارهای زمان و زمان زمان‌بندی تکالیف است.

یک سیستم Real time وقتی پیچیده می‌شود که از یک سری شبکه کامپیوتری به جای یک کامپیوتر استفاده شود.

اجرای تکالیف: این تکالیف نسبتاً از هم مستقل هستند و به صورت دوره‌ای پردازش می‌شوند می‌توان تکالیف را به صورت آشکار زمان‌بندی کرد و فرکانس و پردازش آن یعنی پردازش اطلاعات با فرکانس‌های مختلف صورت می‌گیرد، refrence اصلی اینگونه از آن استفاده می‌شود.

یک قاب اطلاعات بعنوان فاصله تنظیم وقت به صورت پایه به استفاده از مان سنج تعریف می‌شود. در هر قاب به هر تکلیفی داده می‌شود و هر تکلیف در زمان خود اجرا شده و دسترس انحصاری به تمام داده‌ها به صورت سراسری دارد. تربیتی که با آن تکالیف اجراء می‌شود جهت همزمان کردن ارسال اطلاعات بین آنها بکار می‌رود.

تکالیفی که به صورت اکشن عمل می‌شود در فاصله زمانی بین پایان آخرین شکاف و شروع قاب‌بندی فعال می‌شود (دیاگرام انجام تکالیف مدنظر است) که برای تشخیص رویدارها می‌توان از مکانیزم وقفه یا روش رویداری (دوره‌ای) استفاده کرد. یک سری نرم افزار موسوم به زمان‌بندی رویداد وضعیت وسایل جانبی را آزمایش کرده و آن را برای حالتی که مورد نیاز تعریف شده نصب کرده‌اند بررسی می‌کنند به شکل زیر نگاه کنید.



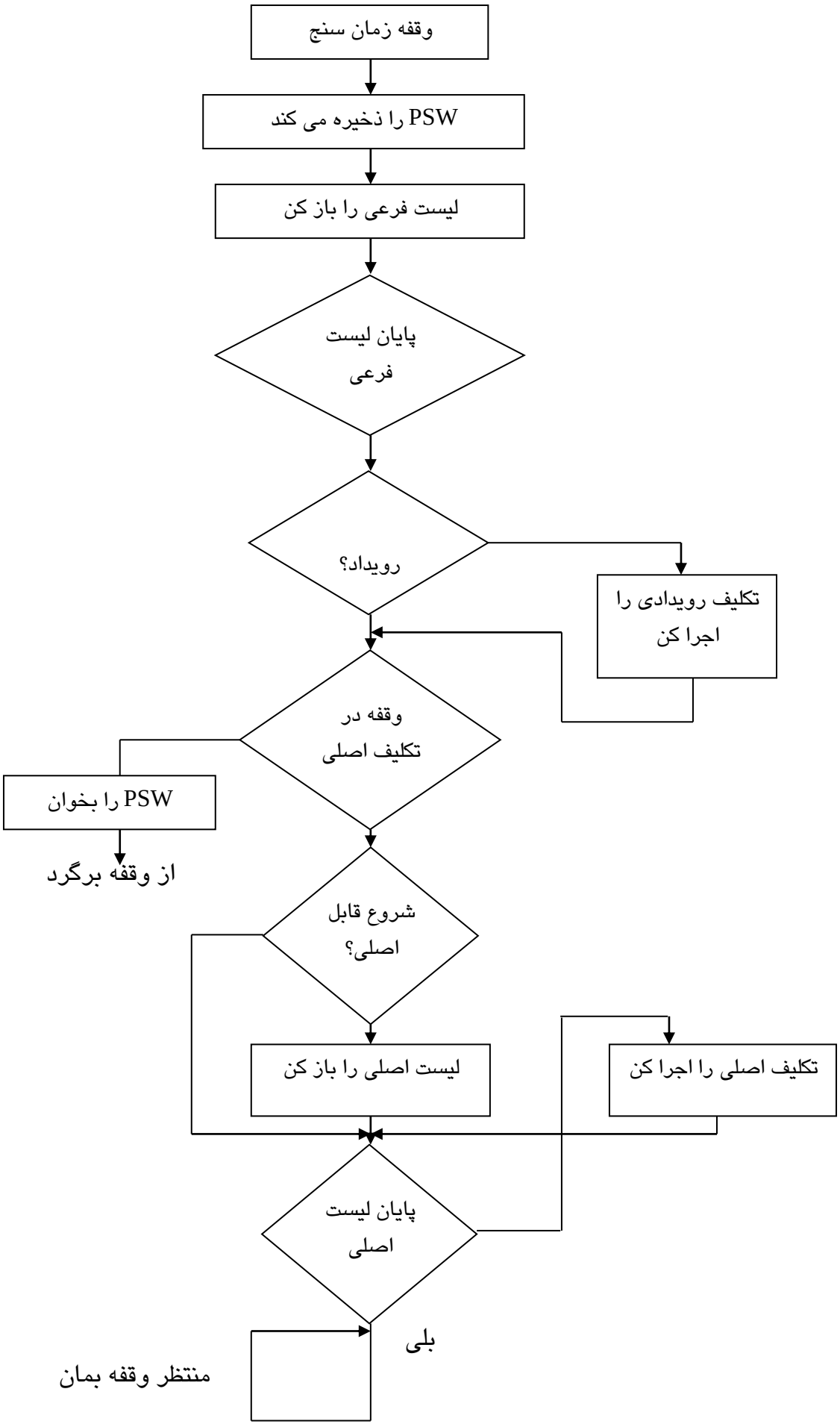
اجرای همگام تکالیف دوره‌یی و رویدادی

به شکل نگاه کنید، تعدادی از قاب‌های در یک ساختار چند سطحی نشان داده شده‌اند که قاب‌های سطح پائین minor frame و قاب‌های بزرگتر major frame هستند در این مثال سطحی شکافی از یکی از قاب‌های فرعی در قاب اصلی نسبت داده می‌شود و در صورتی که تکلیفش بیش از حد بزرگ باشد آن را به تکالیف کوچکتر فرعی تقسیم کرده برروی چند قاب می‌نشاند.

روش اجرای به هنگام تکالیف:

چون اختصاص قاب‌ها می‌تواند پیچیده باشد از یک برنامه اجرائی متمرکز استفاده می‌کنند به این صورت که یک لیستی که از قبل تشکیل شده تعیین می‌کند که کدام تکلیف در کدام قابی فرا خوانده شود و تقسیم تکالیف با فرکانس پائین‌تر را تکالیف فرعی می‌سپارد در ضمن می‌تواند به صورت خودکار ساختار اصلی یک برنامه چرخه‌دار به صورت دو سطحی یا چند سطحی با تقسیم‌بندی خود اجرا کند. دیاگرام زیر ساختار یک برنامه هنگام دو سطحی با استفاده از لیست اصلی و فرعی و یک PSW (کلمه وضعیت اشاره به محتویات شمارنده، برنامه و اطلاعات مورد نیاز است) استفاده کند.

این PSW زمینه تکالیف زیر است،نکته:این الگوریتم در سیستم real time و سیستم عاملIRMX88استفاده می‌شود.


وقفه زمان سنج
PSW را ذخیره می کند
لیست فرعی را باز کن
پایان لیست فرعی
رویداد؟
تکلیف رویدادی را اجرا کن
وقفه در تکلیف اصلی
PSW را بخوان
از وقفه برگرد
شروع قابل اصلی؟
لیست اصلی را باز کن
تکلیف اصلی را اجرا کن
پایان لیست اصلی
بلی
منتظر وقفه بمان

سیستم‌های چند تکلیفی:

استاندارد CCITT چرخه رنگی یک تکلیف را با استفاده از Task state (حالت تکالیف با استفاده از مفهوم تکالیف اجرا می‌کند.

۱- تکلیف بلادرنگ در طول حیات خود یکی از چهار حالت زیر دارد:

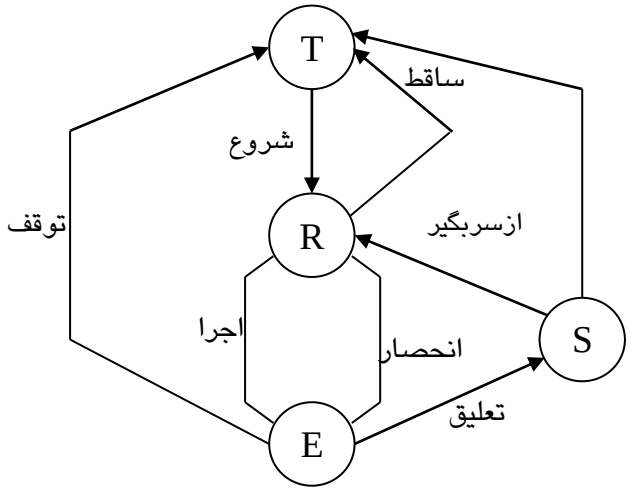
۱. حالت تمام شده (treat): در این حالت تکالیف در لیستی که بوسیله یک سیستم عامل بلادرنگ حفظ می‌شود ثبت شده و وارد می‌گردد.

۲. حالت آماده (Readied): در این حالت دستورات آماده اجرا بوده منتظر پرسسور می‌شوند این به معنی این است که تمام شرایط برای شروع کار حاضر شده به استثنای cpu که مشغول بوده در حال اجرای تکالیف دیگر است.

۳. حالت اجراء (Running): در این حالت پردازنده تکالیفی که به او اختصاص یافته است در حال اجرا دارد و کارکرد آن باعث بوجود آمدن گرما شده که باید رفع شود.

سیستم‌های معلق:

در این حالت تکلیف منتظر اختصاص یک منبع و وقوع یک رویداد با انقضای زمانی معین است. نمودار زیر گذر حالت تکلیف و عملیات است که بیان می‌کند فرق اساسی بین حالت task state و suspended (معلق) را چگونه درک کنیم.



اولیت اجرای تکالیف در سیستم‌ها بلادرنگ:

این که بالاترین ارجحیت را در سیستم‌های کنترلی دارد در جهت اجرای تکالیف اجراء می‌شود. این حالت‌ها توسط رویدادهای شروع می‌شود که در داخل سیستم کامپیوتری رخ می‌دهد. استاندارد **CCITT** این اولیت‌ها را به گروه‌های زیر تقسیم‌بندی کرده است برای اجرای تکالیف:

۱- رویدادهای خارجی (External Event):

مطابق با شرایط در کامپیوترها معمولاً وقفه‌های در آنها ایجاد می‌شود مثل وقفه کارکرد سنسور آستانه به صورت دو حالته این وقفه‌ها در حد میلیونم ثانیه است اما وقتی چند تا از آنها با هم جمع می‌شود عددی بزرگ را شامل می‌شود.

۲- رویدادهای زمانی: time ring Event:

این رویدادها انقضای فواصل زمانی را نشان می‌دهد که با **Dellay** (Δt) مشخص می‌شود. این رویدادها معمولاً توسط اداره کننده زمان سیستم بوجود می‌آید که در کارکرد سیستم تأثیر گذار است.

۳- رویدادهای داخلی: Internals Event:

معرف خطاها و **Error**ها در برنامه هستند این رویدادها با برنامه‌های سخت افزاری مثل وقفه عددها بر صفر ایجاد می‌شوند و نیز اعداد اعشاری زیادی دارند. در سیستم‌های **Real** رویدادهای داخلی زیادی دارند.

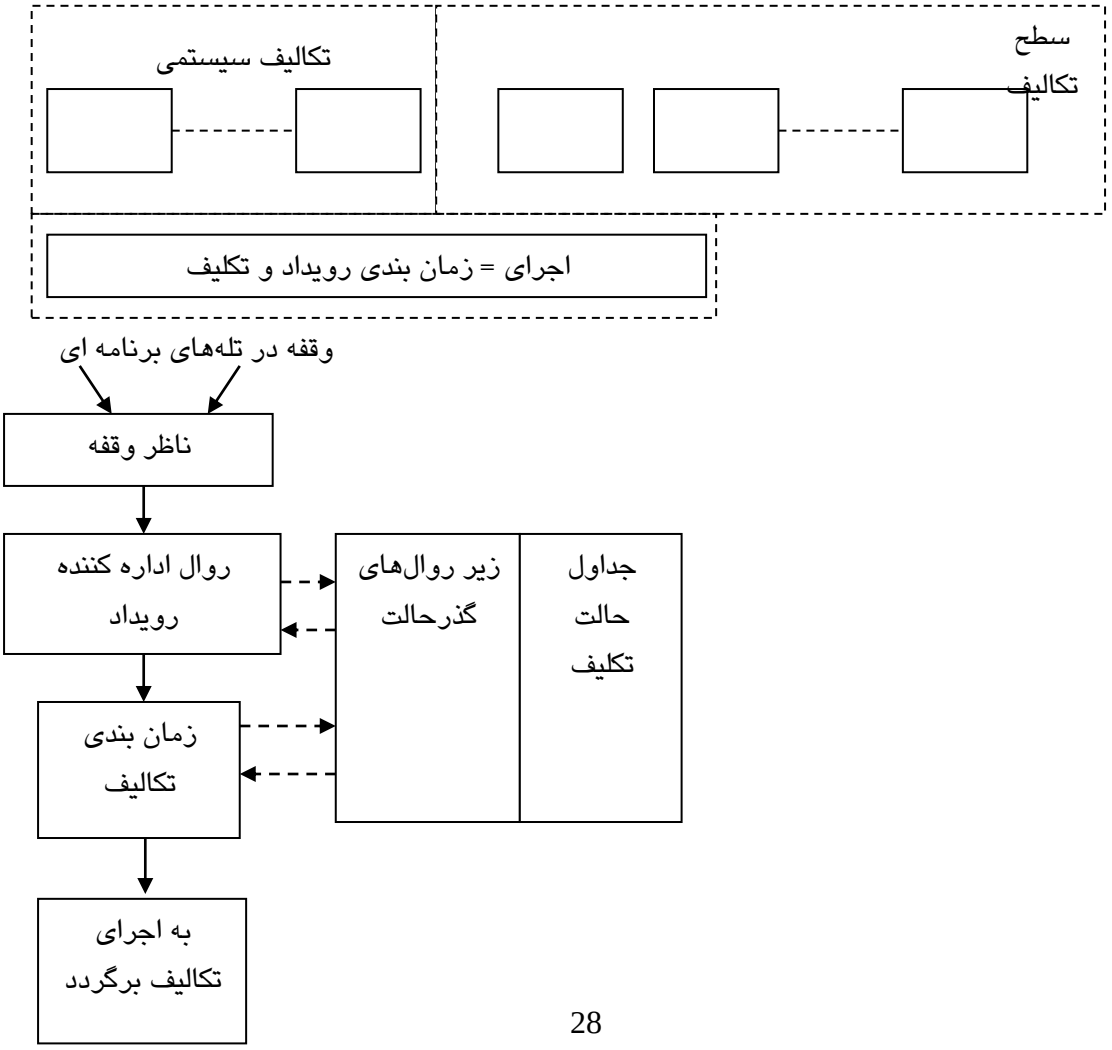
رویدادهای برنامه‌ای:

معرف شرایط خاص می‌باشند که درون یک تکلیف هنگام اجراء، اجراء می‌شوند چنین رویدادهایی درون سیستم عامل توسط تله‌های برنامه‌ای یا فراخوان‌ها به زیر برنامه‌هایی

که توسط سیستم عامل انجام می‌شود، انجام می‌شود. مثل باز کردن یک نرم افزار کوچک‌تر در دل یک نرم افزار بزرگتر.

سیستم‌عامل چند تکلیفی (multitedeing operating)

یک برنامه اجرایی است و هسته سیستم می‌باشد. مسئول اداره تکلیف در رویدادهاست. مدیریت حافظه را برعهده دارد و واحد ناظر وقفه یا interrupt در داخل خود دارد انجام عملیات اداره رویداد را در Event handler را برعهده دارد. در قسمت تحتانی برنامه مجموعه‌ای از جداول تکلیفی را داراست که به آن جداول حالت تکلیف TST می‌گویند که حاوی داده‌های تشریح کننده حالت تکلیف و اولویت‌های منابع است. به دو شکل زیر نگاه کنید یکی مدل سیستم عامل قید تکلیفی را به حالت Real time نشان می‌دهد و دیگری برنامه اجرای چند تکلیفی را نشان می‌دهد.

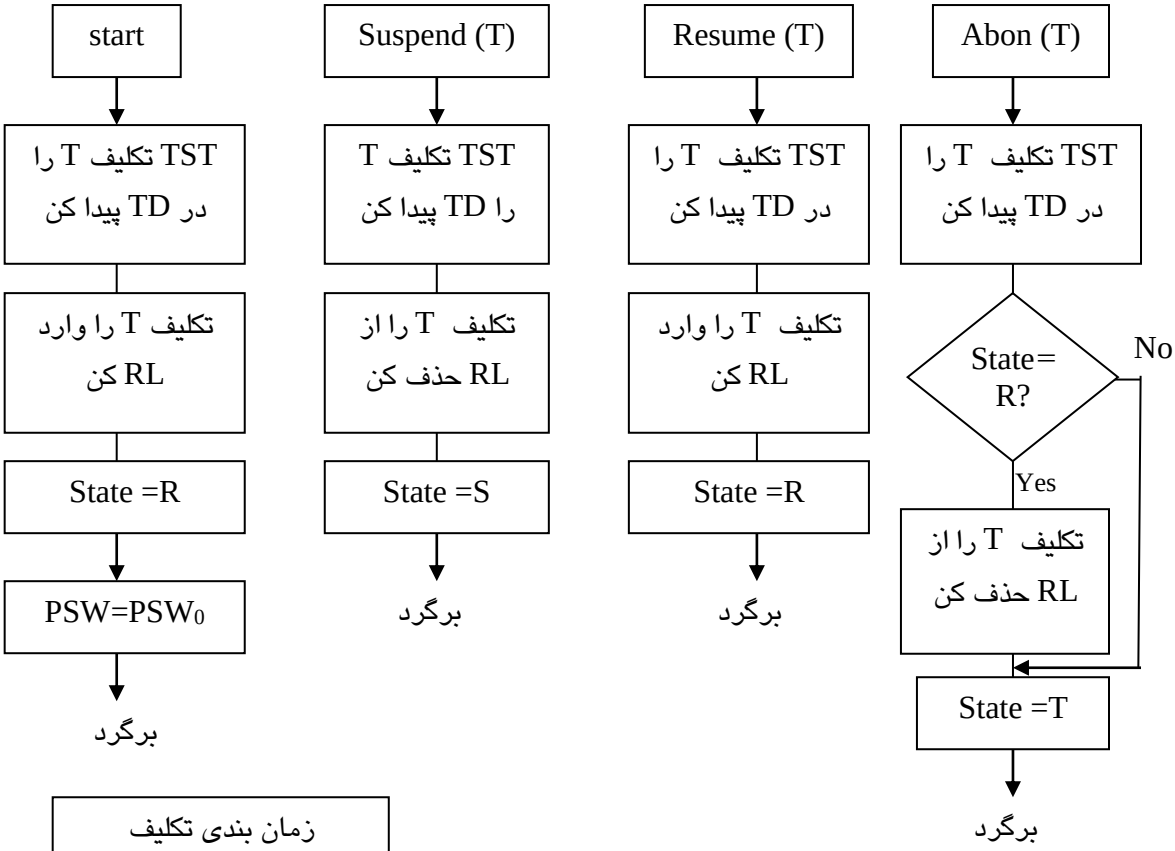


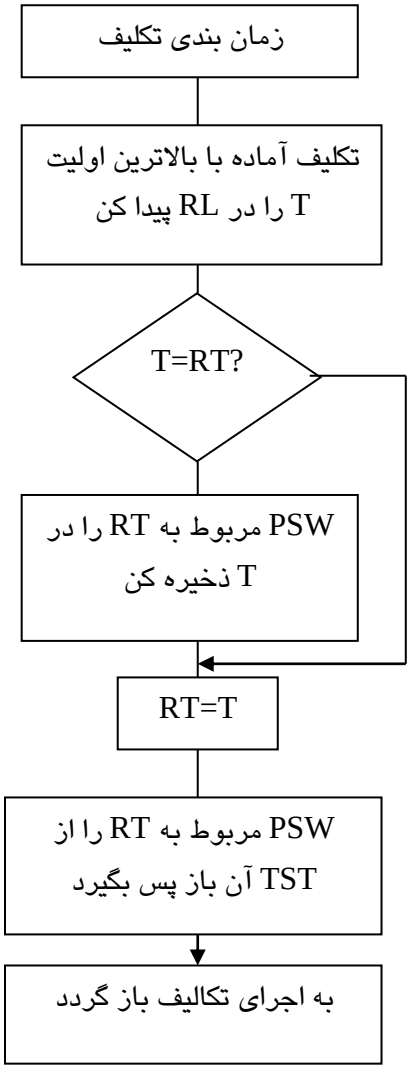
عملیات گذر حالت:

این عملیات را می‌توان برحسب پردازش داده‌های ذخیره شده در جداول حالت تکلیف تشریح نمود این الگوریتم برنامه‌ای را برحسب لیست‌هایی مرتب می‌کنند که این لیست‌ها عبارتند از:

۱- فهرست تکلیف (Task to directory): این فهرست شامل تمام تکالیف ثبت شده در سیستم است و این امکان را بوجود می‌آورد که تمام تکالیف را به آدرس‌های جداول حالت ترجمه نمائیم.

۲- لیست آماده Ready lost (RL): این لیست تکالیفی را تشکیل می‌دهد که آماده اجراء هستند. سایر لیست‌هایی که وجود دارند غالباً زیر مجموعه TD و RL هستند. در یک سیستم TST ذخیره شونده، اولویت‌هایی PWS و PSW جاری را در ۴ برنامه شروع، ثابت، تعلیق و از سرگیری تشریح می‌کنند نشان می‌دهد بعنوان مثال عمل ثابت شدن را برای تمام اعمالی اختصاص داده که باید از عمل بایستند در صورتی که قرار باشد یک تکنیک به صورت پویا در دستگاه کار کند متن برنامه، تکلیف را از ابتدا از حافظه دستگاه به بخش اصلی بار گذاری می‌کند. الگوریتم‌های زیر حالت تکلیف را برای تمامی ۴ حالت شروع، ثابت، تعلیق و از سرگیری به اضافه زمان تکلیف نشان می‌دهد.

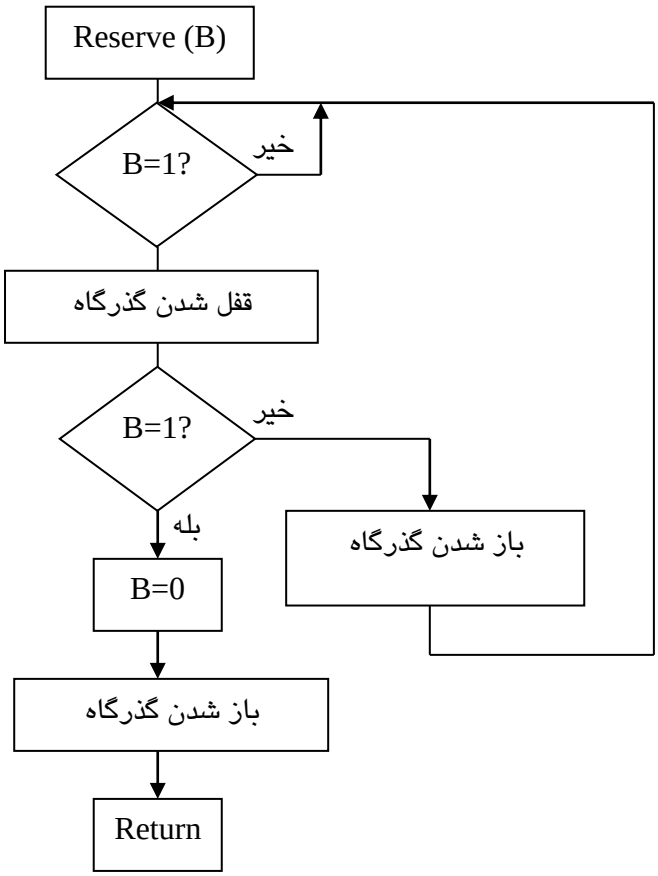

start
TST تکلیف T را
در TD پیدا کن
تکلیف T را وارد
RL کن
State =R
PSW=PSW0
برگرد
Suspend (T)
TST تکلیف T
را TD پیدا کن
تکلیف T را از
RL حذف کن
State =S
برگرد
Resume (T)
TST تکلیف T را
در TD پیدا کن
تکلیف T را وارد
RL کن
State =R
برگرد
Abon (T)
TST تکلیف T را
در TD پیدا کن
State=
R?
No
Yes
تکلیف T را از
RL حذف کن
State =T
برگرد

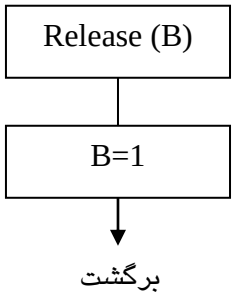

زمان بندی تکلیف
تکلیف آماده با بالاترین اولیت
T را در RL پیدا کن
T=RT?
PSW مربوط به RT را در
T ذخیره کن
RT=T
PSW مربوط به RT را از
TST آن باز پس بگیرد
به اجرای تکالیف باز گردد

واحد مدیریت حافظه:

حافظه بصورت کاربردی و نظارتی در سیستم‌های Real time استفاده می‌شود به دستگاهی که هر دو این المان را دارند دستگاه حافظه ممتاز می‌گویند در ساده‌ترین حالت دستیابی انحصاری به داده‌ها به صورت مشترک، استفاده از گذرگاه سیستم است که با قفل کردن گذر و باز کردن آن، عملیات انجام می‌شود. این روش کمی سیستم را دچار اختلال می‌کند. روش جایگزین، استفاده از مفهوم نشان دودویی (requite) است. به طور انتخابی در حافظه دخالت کرده و نشان می‌دهد که آیا سیستم مشغول است یا خیر این الگوریتم شامل عمل B بوده که resave را نشان می‌دهد.

وقتی B=0 است یعنی پردازنده مشغول بوده و در حالت تعلیق است. عمل release منبع را آزاد کرده و B را مساوی 1 می‌کند. الگوریتم مربوط به عمل resave نوع عمل فعال انتظار بوده و از مکانیزم سخت افزار انحصاری حفظ گذرگاه استفاده می‌کند. بهتر است به عملیات نشان بری فعال در واحد مدیریت حافظه نگاه کنیم:





اداره کردن رویدادها و زمان‌ها:

تکالیفی که در یک سیستم بی‌درنگ، اجرا می‌شود بطور دائم بوده و دارای حلقه‌ای بی‌پایان است. دوره تناوب اجرای حلقه‌ها قابل تعریف شدن است و مکانیزم‌های لازم برای اندازه‌گیری و سیگنالینگ بوسیله سیستم عامل فراهم می‌شود از ابزارهای سیگنال دهی زمان عمل Delay کردن است این delay بصورت (Δt) سبب تعلیق یافتن تکلیف برای

یک وره زمانی (Δt) است مشکلی که در استاندارد **DIN** برای این سیستم وجود دارد اضافه کردن (Δt) به زمان اجرای تکلیف است که نمی‌توان دقیقاً آن را تعریف کرد برای این منظور استاندارد اینگونه بیان کرده است:

بهترین راه این است که تکالیف را در فواصل زمانی منظم با استفاده از قالب‌های اطلاعاتی تعریف کند.

به نحوی که هر قاب قابلیت از سرگرفتن تکلیف را داشته باشد زمان تعلیق یا **Time out** حداکثر زمانی است که می‌توان تکلیفی را در حالت تعلیق نگهداشت. سیگنال مربوط به آن (Δt) طول زمان نامیده می‌شود. سیستم عامل **Real** می‌توان لیست انجام تکالیفی را در حافظه دستگاه برای فواصل زمانی معین که به تعلیق در آمده‌اند نگهدارد از این زمان سنجی برای دو پردازش اساسی $\Delta cycle = \Delta delay$ استفاده کرد.

استاندارد **CCITT** مکانیزم تنظیم وقت سیستم‌های **Real** را اینگونه انجام می‌دهد.

۱- رویدادهای خارجی **External events**

۲- رویدادهای داخلی **Internal events**

نکته: در سیستم‌های عامل **Real** رویدادهای داخلی توسط سیستم عامل کنترل می‌شود که در بحث ما نمی‌گنجد ولی رویدادهای خارجی اینگونه اداره می‌شوند که مانند سبک سیستم وقه برنامه‌نویسی بوده و دارای سطرهای مختلف است.

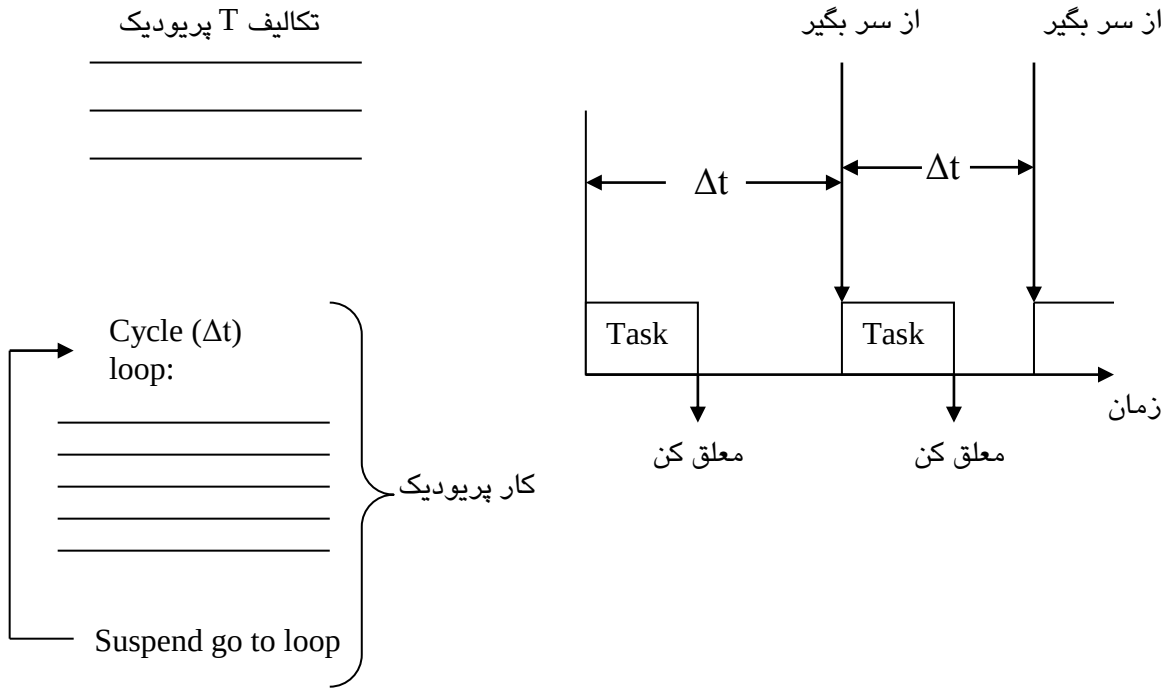
Connect: وقفه‌ای که با رمزهای **INT** معرفی می‌شود با رویدادهای **EV (event)** معرفی می‌گردد. این کار به ناظر وقفه اطلاع می‌دهد تا اطلاعات مربوطه به وقوع وقفه به واحد اداره کننده رویداد برساند.

Enable EV: وقفه‌ای که مطابق با رویداد **EV** می‌باشد فعال است و در صورتی که رویداد واقع شود واحد اداره کننده رویداد آن را مورد توجه قرار می‌دهد.

Awaits EV: **تکالیفی که این عمل را انجام می‌دهد اجرای خود را به تعلیق در می‌آورد و منتظر می‌ماند تا دستور به اجرا برسد.**

Disconnect EV: **در حقیقت تطابق بین رویداد و وقفه است. در این حالت سعی می‌شود به قطع ارتباط بسته‌های عملیاتی همه توسط سیستم عامل real اجرا شود.**

شکل صفحه بعد اجرای دوره‌ای تکالیف را در یک پریودیک نشان می‌دهد.



سیستم‌های عامل توزیع شده:

سیستم‌های عامل شبکه‌های کامپیوتری را در سیستم‌های Real time کنترل می‌کنند. چندین پردازنده را با هم یا تک تک کنترل می‌کنند. در هر کامپیوتر، اینکه فایل‌ها در کجا قرار دارند و اختصاص منابع جهت ذخیره‌سازی آنها متوسط سیستم عامل انجام می‌شود که به آن سیستم عامل توزیع شده می‌گوید. کامپیوترهایی که یک سیستم عامل توزیع شده را تشکیل می‌دهند دارای حافظه اصلی مشترک نیستند نظیر صندوق‌هایی پست عموماً غیرقابل استفاده هستند مدل مرجع ISO در خصوص LANهای سریع از سیستم‌های اتصال‌گرا استفاده کرده و با این کار ارتباط بین کامپیوترها را کامل می‌کند در دنیا از چند سیستم عامل بی‌درنگ استفاده می‌شود که جهت تنظیم وقت سیگنال‌های خود از GPS (general position system) استفاده می‌کنند و با دقت 100ns زمان دقیق انجام تکالیف سیستم‌ها را به آن می‌گوید. سیستم‌های عامل IRNX88 ، IRNX86 ، QMX و PORTOS از سیستم‌های عامل بی‌درنگ مورد استفاده در دنیا هستند که همگی آنها از زبان‌های سطح بالا استفاده می‌کنند.

با آرزوی شادکامی و سربلندی