

10・2 制御系のデジタルシミュレーション

制御系の過渡応答は、前節で説明したように、総括伝達関数に入力を掛けて、これを逆ラプラス変換すれば求まる。しかし、制御器のパラメータを種々変化させて、それぞれの応答を求める作業はかなり大変である。そこで、制御系のデジタルシミュレーションによって、直接過渡応答を計算することは、制御系の特性を調べるための有力な手段となる。

これまでに、先の第4章で、積分要素、1次遅れ要素、2次遅れ要素およびむだ時間要素のシミュレーション手法について説明した。ここでは、はじめに、PID-制御器のシミュレーションプログラムを示す。PID-制御器のブロック線図は、図10・5の下段に示すように三つの要素に分けて示される。

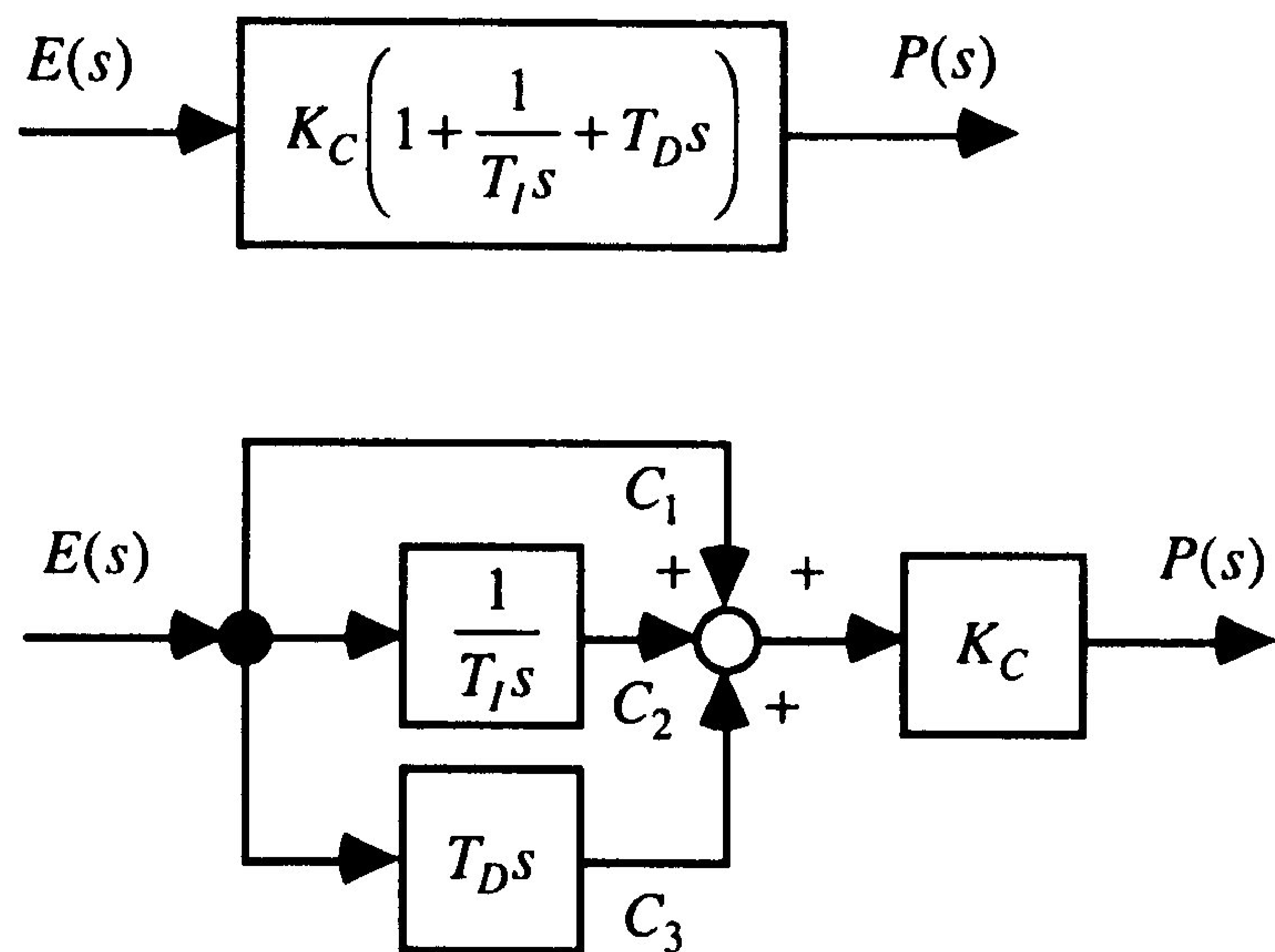


図 10・5 PID-制御器のブロック線図

ここで、 $E(s)$ 対 C_2 の関係は先に示した積分要素である。 $E(s)$ 対 C_3 の関係は微分要素で、これは時間領域では次式のように表すことができる。

$$c_3(t) = T_D \frac{de(t)}{dt} \quad (10.21)$$

$$\approx T_D \{e(t) - e(t - \Delta t)\} / \Delta t$$

この関係をシミュレーションするには、時間 t における入力と 1 時間刻み前の入力との差を、時間刻み幅 Δt で割って、定数 T_D を乗ずればよい。

入力 $e(t)$ を読み込みながら、出力 $p(t)$ を計算するプログラムの例を以下に示す。

プログラムリスト：5

```

1000      REM PID-CONTROLLER
1010      CLEAR: WAIT
1020      INPUT "KC="; KC
1030      INPUT "TI="; TI
1040      INPUT "TD="; TD
1050      INPUT "DT="; DT
1060      INPUT "TE="; TE
1100      REM
1110      FOR TM=0 TO TE STEP DT
1120      PRINT "TM="; TM
1130      INPUT " E="; E
1140      C1=E
1150      C2=C2+(1/TI)*E*DT
1160      C3=TD*(E - EB)/DT: EB=E
1170      IF TM=0 THEN C2=0: C3=0
1180      P=KC*(C1+C2+C3)
1190      PRINT " P="; P
1200      NEXT TM
1210      END

```

- ・行番号1160は微分動作を表している。微分動作は、現在の入力と時間刻み一つ前の入力 (EB) との差を DT で割り、微分時間を乗じたものである。この行の後半では、次回の計算のために、現在の入力を EB に置き直している。この新しい EB が、次回では一時間刻み前の値となる。
- ・行番号1170は IF 文と呼ばれ、 TM が 0 のときには、 $C2$, $C3$ を 0 としてから次の1180行を実行する。 TM が 0 でないときには、 $C2$ を1150行で、 $C3$ を1160行で計算された値として1180行を実行する。この行の働きは、入力初期に生じる微分動作の *derivative kick* (p104) を除くための処置で、同時に、 TM が 0 のときには、積分動作の $C2$ もゼロとした。

フィードバック制御系のシミュレーションの一例として、プロセスは最も単純な1次遅れ要素で、検出部はむだ時間要素で、制御器にはPID-調節計を使用する系を考える。そのブロック線図を図10・6に示す。

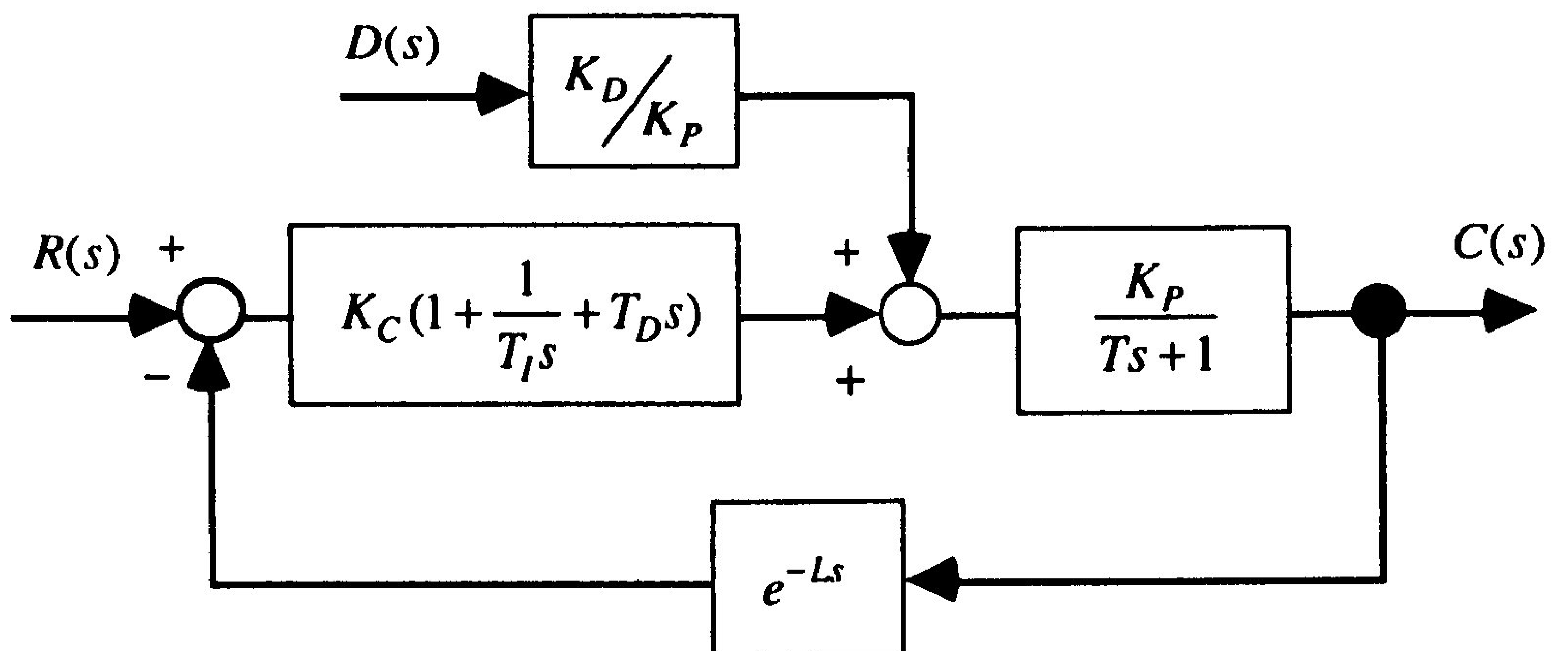


図 10・6 フィードバック制御系 (I) のブロック線図

ここで、外乱がプロセス本体の入力部に挿入されている。通常、プロセス本体の伝達関数と外乱の伝達関数とは、その遅れが同じでゲインのみが違っている（伝達関数の定数部のみが異なる）ことが多い。[p79, (7.2)式のように] そこで、ブロック線図の等価変換により、外乱挿入部を図10.6のように書き直すことができる。

上記の制御系で、設定値あるいは外乱をステップ状に変化させたとき、出力を求めるプログラムを以下に示す。

プログラムリスト：6

```

2000 REM FEEDBACK CONTROL SYSTEM-1
2010 CLEAR: WAIT: JC=11
2020 INPUT" DT=";DT
2030 INPUT" TE=";TE
2040 REM PROCESS CONSTANTS
2050 INPUT" KP=";KP
2060 INPUT" KD=";KD
2070 INPUT" T=";T
2080 INPUT" L=";L
2090 NL=INT(L/DT+0.5)
2100 DIM D(NL)
2110 REM PID PARAMETERS
2120 INPUT" KC=";KC
2130 INPUT" TI=";TI
2140 INPUT" TD=";TD
2150 REM INPUT/SET POINT CHANGE
2160 INPUT" R=";R
2170 REM INPUT/DISTURBANCE
2180 INPUT" DS=";DS
3000 REM CONTROL SYSTEM RESPONSE
3010 PRINT"TM=";TM
3020 PRINT" C=";C

```

```
3030 FOR TM=0 TO TE STEP DT
3040 J=J+1
3050 IF J<JC THEN 3090
3060 PRINT"TM=";TM
3070 PRINT" C=";C
3080 J=1
3090 REM COMPARATOR / IN R,M, OUT E /
3100 E=R-M
3110 REM PID CONTROLLER /IN E, OUT P/
3120 C1=E
3130 C2=C2+(1/TI)*E*DT
3140 C3=TD*(E-EB)/DT
3150 EB=E
3160 IF TM=0 THEN C2=0: C3=0
3170 P=KC*(C1+C2+C3)
3180 REM DISTURBANCE IN
3190 X=P+(KD/KP)*DS
3200 REM 1ST ORDER SYSTEM /IN X, OUT Y/
3210 U=(1/T)*(KP*X-Y)
3220 Y=Y+U*DT
3230 C=Y
3240 REM TIME LAG / IN Y, OUT M/
3250 D(0)=Y: M=D(NL)
3260 IF NL=0 THEN 3300
3270 FOR I=NL TO 1 STEP -1
3280 D(I)=D(I-1)
3290 NEXT I
3300 REM NL=0 OUT
3310 NEXT TM
3320 END
```

ここで、プログラムリスト：6について説明する。

- ・行番号2050～2080でプロセスのゲイン，時定数，むだ時間を読み込む。
- ・行番号2090，2100でむだ時間用の配列を宣言する。
- ・行番号2120～2140でPIDパラメータを読み込む。
- ・行番号2160，2180で設定値および外乱を入力する。
- ・行番号3030は，行番号3310と対応し，この間で TM を1刻み DT ずつ増加させていく。行番号3040で，時間が1刻み増加するたび J を1ずつ増やしていく。
- ・行番号3050では， J が JC より小さいときには行番号3090に跳ぶ。 J が JC に等しいか，大きくなると，行番号3060以下で TM ， C を表示し， J を1に戻して，行番号3090に続く。これは， TM ， C の表示を $(JC-1)$ 回毎に行うよう工夫したものである。なお， JC の値は行番号2010で与えられている。
- ・行番号3100で設定値と検出値との差を求める。
- ・行番号3120～3170でPID制御器の動作をシミュレーションしている。
- ・行番号3190で制御器の出力と外乱との和を求める。
- ・行番号3210～3230で1次遅れ要素の応答を求める。
- ・行番号3250～3300でむだ時間の応答を求める。行番号3260では， $L=0$ のとき，すなわち NL が0のとき行番号3300に跳ぶよう処置している。

次のシミュレーションでは，プロセスの伝達関数として1次遅れ要素が2個連なった2次遅れ要素で，検出部も1次遅れ要素で与えられる制御系を考える．ここで，制御器としてPID-調節計を使用する．この制御系のブロック線図を図10・7に示す．

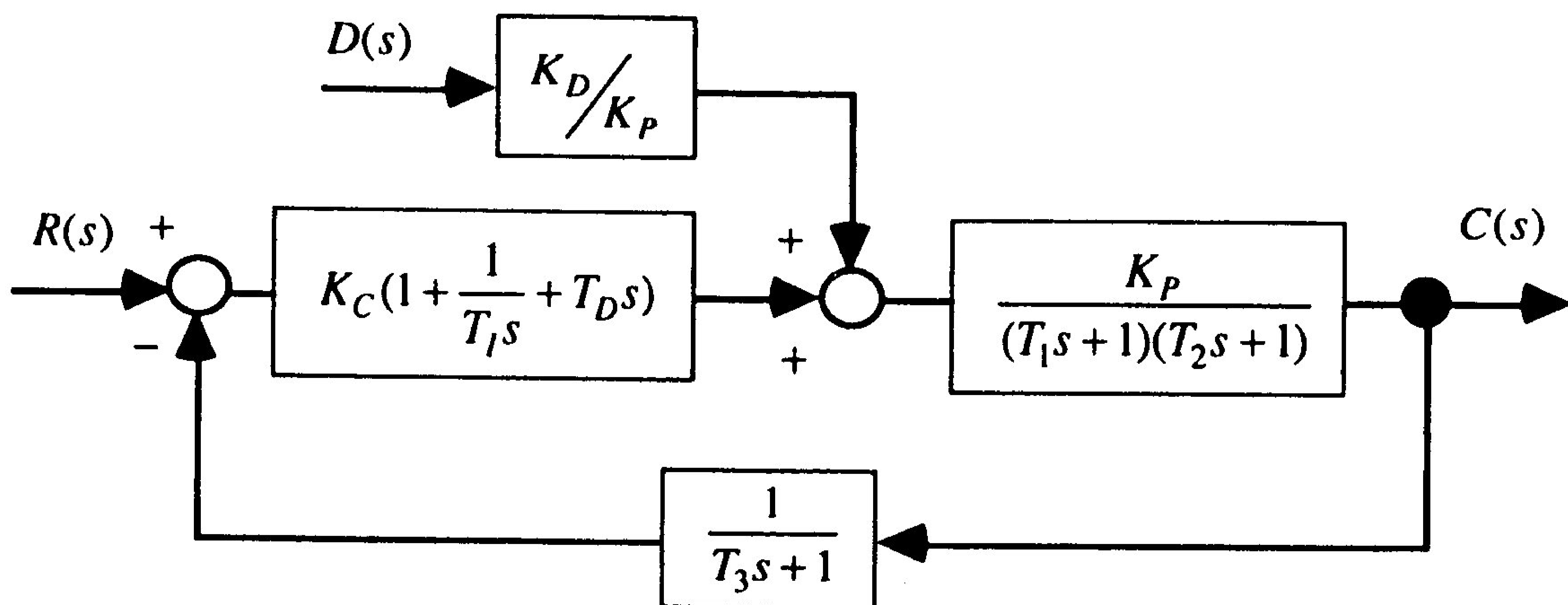


図 10・7 フィードバック制御系 (II) のブロック線図

上記の制御系で，外乱あるいは設定値のステップ変化に対する応答を求めるプログラムを以下に示す．

プログラムリスト：7

```

4000 REM FEEDBACK CONTROL SYSTEM-2
4010 CLEAR: WAIT: JC=11
4020 INPUT" DT=";DT
4030 INPUT" TE=";TE
4040 REM PROCESS CONSTANTS
4050 INPUT" KP=";KP
4060 INPUT" KD=";KD
4070 INPUT" T1=";T1
4080 INPUT" T2=";T2
4090 INPUT" T3=";T3

```

```
4100 REM PID PARAMETERS
4110 INPUT" KC=";KC
4120 INPUT" TI=";TI
4130 INPUT" TD=";TD
4140 REM INPUT/SET POINT CHANGE
4150 INPUT" R=";R
4160 REM INPUT/DISTURBANCE
4170 INPUT" DS=";DS
5000 REM CONTROL SYSTEM RESPONSE
5010 PRINT"TM=";TM
5020 PRINT" C=";C
5030 FOR TM=0 TO TE STEP DT
5040 J=J+1
5050 IF J<JC THEN 5090
5060 PRINT"TM=";TM
5070 PRINT" C=";C
5080 J=1
5090 REM COMPARATOR / IN R,M, OUT E /
5100 E=R-M
5110 REM PID CONTROLLER /IN E, OUT P/
5120 C1=E
5130 C2=C2+(1/TI)*E*DT
5140 C3=TD*(E-EB)/DT
5150 EB=E
5160 IF TM=0 THEN C2=0: C3=0
5170 P=KC*(C1+C2+C3)
5180 REM DISTURBANCE IN
5190 X=P+(KD/KP)*DS
5200 REM 2ND ORDER PROCESS /IN X, OUT Y/
5210 U1=(1/T1)*(KP*X-Y1)
5220 Y1=Y1+U1*DT
5230 U2=(1/T2)*(Y1-Y)
```



```

5240 Y=Y+U2*DT
5250 C=Y
5260 REM 1ST ORDER ELEMENT/ IN Y, OUT M/
5270 U3=(1/T3)*(Y-M)
5280 M=M+U3*DT
5290 NEXT TM
5300 END

```

ここで、プログラムリスト：7について説明する。

- ・プログラムの構成は、先に示したリスト：6とほとんど同じである。
- ・行番号4050～4090でプロセスのゲイン，時定数，および検出部の時定数を読み込む。
- ・行番号5210～5250でプロセス部の応答を計算している。行番号5210，5220で最初の1次遅れ要素の応答を，行番号5230，5240で次の1次遅れ要素の応答を計算している。
- ・行番号5270，52800で検出部の応答を計算している。

この図10・7のフィードバック制御系（II）について，PID-制御器のパラメータの働きを検討する。ステップ状の外乱に対する制御系の応答を，プログラム7によって数値計算し，その結果を図10・8～10・10に示す。

* 時間刻み幅として，プロセスの最も小さい時定数の1/10～1/20にすると，数値計算に伴う誤差を小さくすることができる。

プロセスの特性および計算条件を以下のように定める。

- ・ プロセス特性： $KP = 1, KD = 1, T1 = 0.5, T2 = 1.0, T3 = 0.2$
- ・ 計算条件*： $DT = 0.01, TE = 6, R = 0, D = 1$

まずはじめに，比例制御のみの制御結果を示す。制御器のパラメー

タを以下のように設定する。

- 計算条件： $K_C = 0, 5, 10, 20$, $T_I = 10^5$, $T_D = 0$

ここで、 $K_C = 0$ の計算はフィードバック制御を行わない、オープンループの結果を示す。積分、微分制御は行わないので、積分、微分時間を上記のような数値にした。P-制御の結果を図10・8に示す。

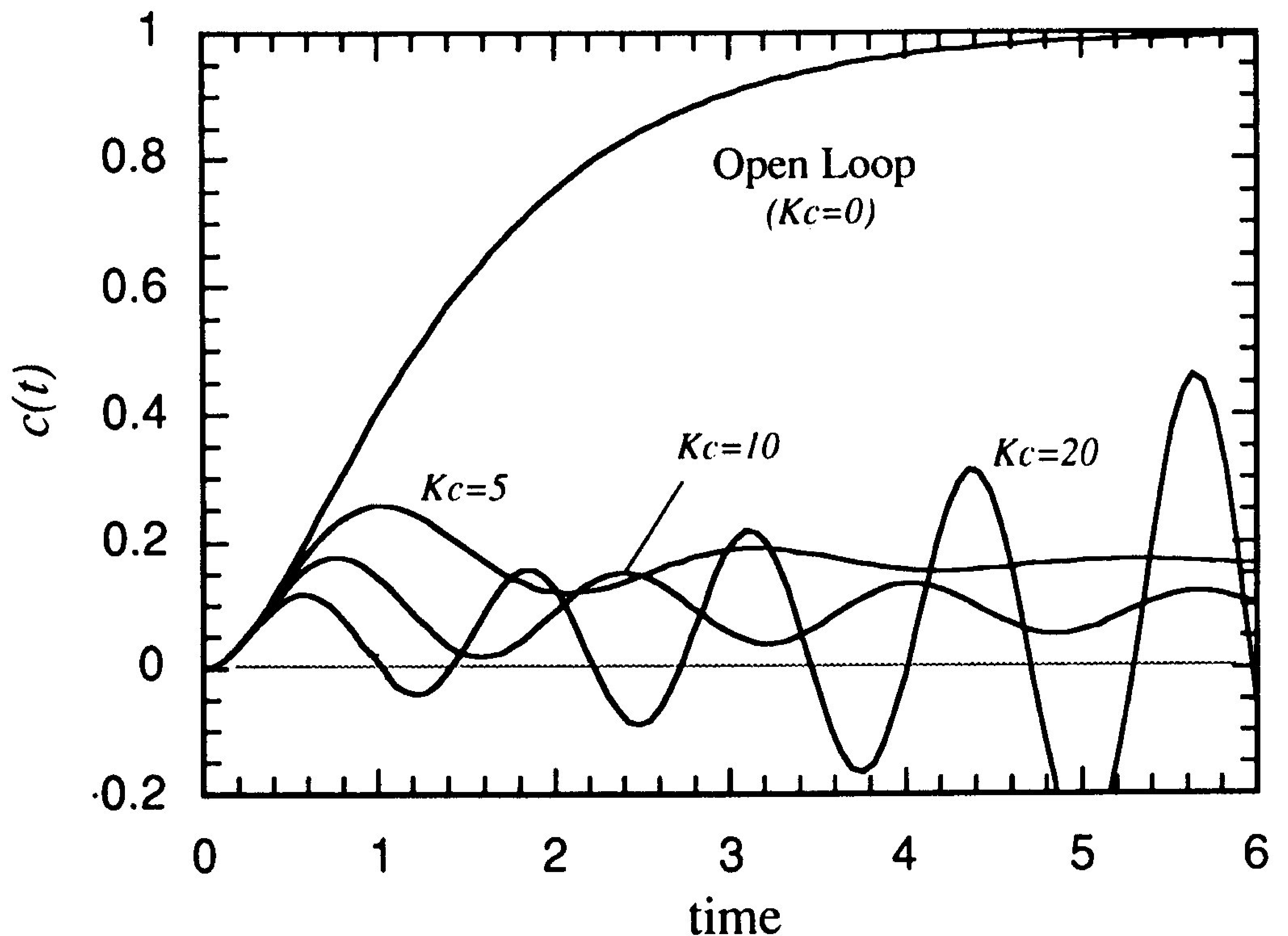


図 10・8 比例制御のステップ応答

この結果を見ると、比例ゲインが大きくなるにつれ、オフセットは小さくなる。しかし、その応答には振動成分が含まれており、 $K_C = 20$ では、その振動が発散して制御系は不安定となる。

次に、比例-積分制御の応答を示す。制御器のパラメータ・積分時間を以下のように変化させた。

- 計算条件： $K_C = 5$, $T_I = 10^5, 2.0, 1.0, 0.5$
外乱に対するステップ応答を図10・9に示す。

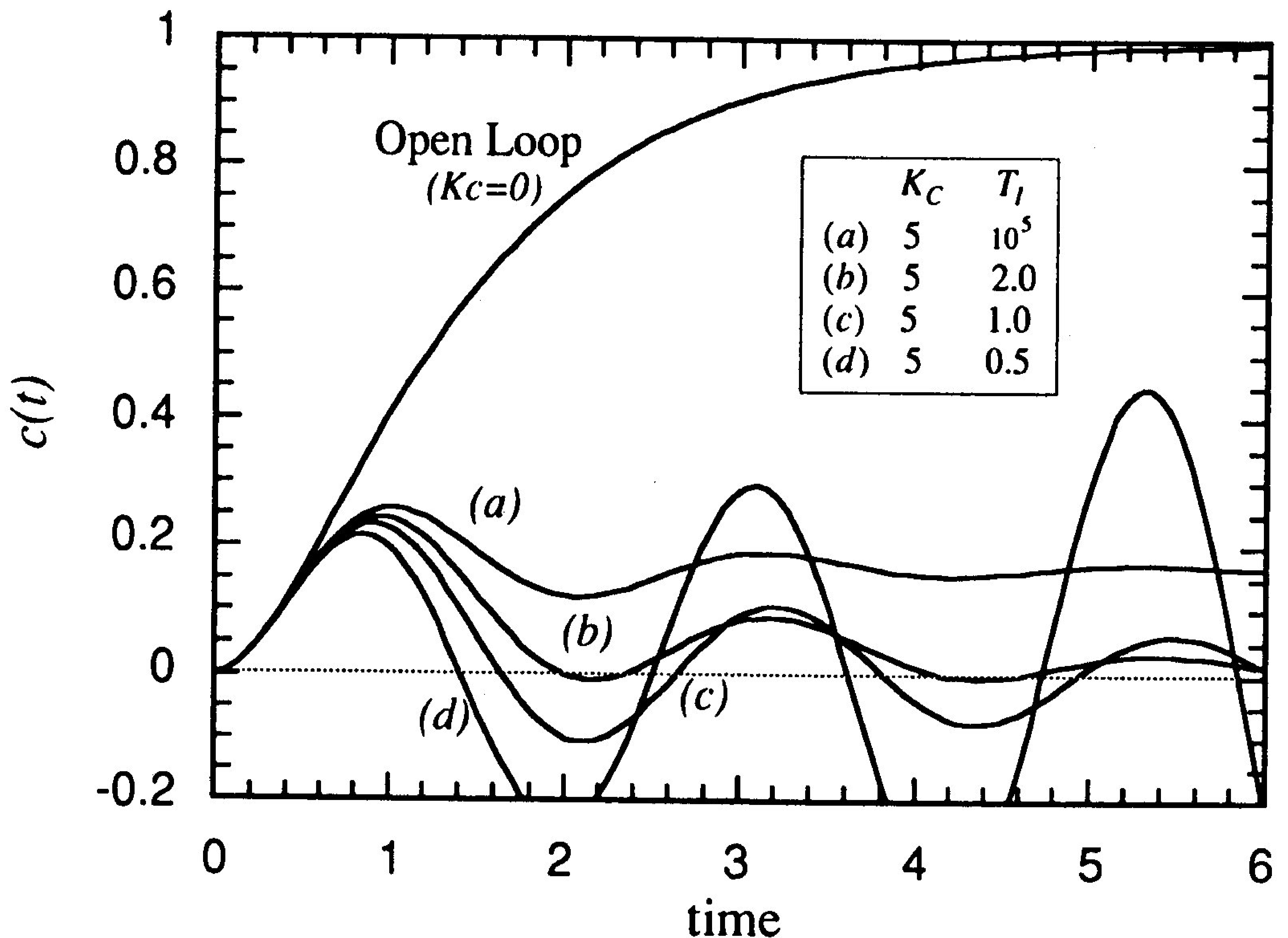


図10・9 比例-積分制御のステップ応答

積分動作を加えることによって、オフセットを取り除くことができた。積分動作を強めるために、積分時間を小さくしていくと、応答はより振動し易くなり、 $T_I = 0.5$ の場合には制御系は不安定となった。一般的に、積分動作を付け加えると、制御系の応答の振動成分が大きくなり、系の安定性は悪くなる。

次に、比例-積分-微分制御の結果を示す。

- 計算条件： $K_C = 5$, $T_I = 1$, $T_D = 0, 0.2, 0.6$

微分時間を変化させて得られた制御系の応答を図10・10に示す。

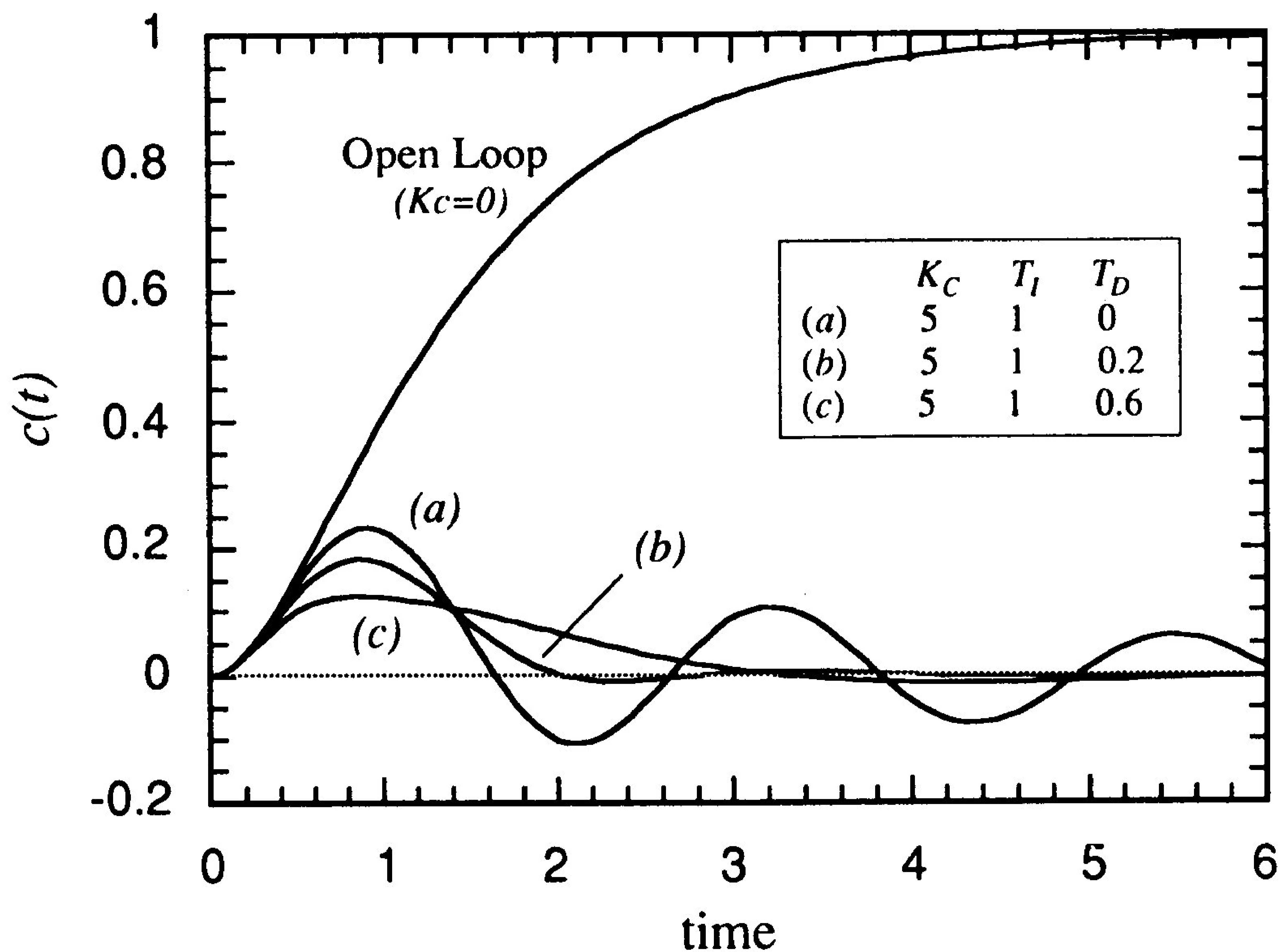


図 10・10 比例-積分-微分制御のステップ応答

上記の応答を見ると、微分動作を付け加えることによって、振動性の応答が穏やかになることがわかる。微分動作を強くしていくと、応答はより穏やかになるが、最終値に収束するまでの時間が長くなる。一般的に云って、微分動作は制御系の安定性を増加させる働きがある。