

Messen Steuern Regeln
Leistungstest von Prozeßrechensystemen
Zeitmessungen

Testfunktion: Suchen eines Bitmusters

DIN
19 242
 Teil 6

Measurement and control; performance tests on process computer systems, time measurement; function tested: detection of a bit pattern

Für den Anwendungsbereich dieser Norm bestehen keine entsprechenden regionalen oder internationalen Normen.

Inhalt

	Seite		Seite
1 Anwendungsbereich und Zweck	1	5.2 Verweilzeit	2
2 Allgemeines	1	6 Testumgebung	2
3 Testfunktion	1	7 Testprotokoll	2
4 Testvorgang	1	Anhang A	3
4.1 Vorbereitung	1	Anhang B	9
4.2 Durchführung	1	Anhang C	15
5 Testergebnisse	2	Zitierte Normen	20
5.1 Belegungszeit des Zentralprozessors (t_{zp})	2	Erläuterungen	20

1 Anwendungsbereich und Zweck

Mit dieser Testfunktion soll die Bearbeitungszeit für das Aufsuchen eines Bitmusters nach dem Testverfahren DIN 19 242 Teil 1 gemessen werden.

Um die Programmierung der Testfunktionen zu erleichtern, sind in den Anhängen A, B und C Programmbeispiele in FORTRAN, PEARL und PASCAL beigefügt, die mit geringfügigen Änderungen auf allen Rechnern ablaufen können. Diese Beispiele sind auch beim Beuth Verlag auf Datenträger (z. B. Magnetband) erhältlich.

2 Allgemeines

Zu verarbeitende Bitmuster stimmen in der Regel nicht mit der Verarbeitungseinheit des Prozeßrechensystems überein.

Für den Test wird ein Bitmuster gewählt, das kleiner als 8 Bit ist, damit eine möglichst große Anzahl von verschiedenen Prozeßrechensystemen mit diesen Testprogrammen gemessen werden kann.

3 Testfunktion

Suchen eines vorgegebenen Bitmusters der Länge 5 Bit an den Bitpositionen 2 bis 6 innerhalb einer Verarbeitungseinheit des Zentralprozessors im Hauptspeicher (Beispiele siehe Anhänge).

4 Testvorgang

4.1 Vorbereitung

Im Vorlauf des Testprogrammes wird ein entsprechend großer Hauptspeicherbereich des Prozeßrechensystems mit Bitmustern gefüllt. Diese Bitmuster werden so gewählt, daß sich eine vom Anwender gewünschte Erfolgs- bzw. Mißerfolgsquote beim Suchvorgang ergibt.

Der Wiederholungsfaktor k für die Testfunktion muß so gewählt werden, daß die Testdauer im Minutenbereich (mindestens 1 bis 10 Minuten) liegt.

4.2 Durchführung

Der Zugriff auf die Bitmuster im Hauptspeicher soll abwechselnd von der unteren bzw. oberen Adreßgrenze aus jeweils in entgegengesetzter Adressierungsrichtung erfolgen.

Die entsprechend modifizierte Hauptspeicher-Adresse wird in der Suchfunktion verwendet.

Der Inhalt der adressierten Hauptspeicherzelle wird mit einem vorgegebenen Bitmuster verglichen. Bei Erfolg bzw. Mißerfolg findet eine Verzweigung zur Zählung statt. Zur Suchfunktion gehören außer der obengenannten Adresse noch weitere Parameter:

- die Bitfeldanfangsposition
- die Bitfeldlänge
- das Bitmuster (Inhalt des Bitfeldes).

Fortsetzung Seite 2 bis 20

Deutsche Elektrotechnische Kommission im DIN und VDE (DKE)

Jede Art der Vervielfältigung, auch auszugsweise, nur mit Genehmigung des DIN Deutsches Institut für Normung e. V., Berlin, gestattet.

Um die Zeiten für die Testfunktion zu ermitteln, muß der Testvorgang in zwei Schritten durchgeführt werden:

Schritt 1: Testlauf ohne Suchfunktion

Schritt 2: Testlauf mit Suchfunktion.

5 Testergebnisse

k Wiederholungsfaktor

andere Formelzeichen: siehe DIN 19 242 Teil 1

5.1 Belegungszeit des Zentralprozessors (t_{ZP})

$$t_{ZP} = [(t_{EH2} - t_{AH2}) - (t_{EH1} - t_{AH1})] / k$$

5.2 Verweilzeit

$$t_{VW} = [(t_{EV2} - t_{AV2}) - (t_{EV1} - t_{AV1})] / k$$

6 Testumgebung

Grundkonfiguration nach DIN 19 242 Teil 1/01.87, Abschnitt 7.

7 Testprotokoll

Zum Testergebnis muß die Testumgebung (siehe DIN 19 242 Teil 1) im Protokoll hinreichend genau beschrieben werden, damit der Test reproduzier- und vergleichbar wird.

Die unter Abschnitt 4.1 gewählte Erfolgs- bzw. Mißerfolgsquote muß im Protokoll vermerkt sein.

Die gewählten Parameter sind anzugeben.

Anhang A FORTRAN-Programm

```

C*****
C****
C**** PROGRAMMIER-SPRACHE:...FORTRAN IV
C****
C****
C****
C****
C****
C****
C****
C*****
C****
C**** DER TESTRAHMEN BESTEHT NACH DIN 19242 TEIL 1 AUS DEN
C**** PROGRAMMABSCHNITTEN:
C****
C****      1). VORSPANN (EINLESEN VON PARAMETERN, INITIALISIEREN,
C****              STARTEN)
C****      2). BEREICH FUER TESTFUNKTION BZW. TESTHILFSFUNKTION
C****              (FUNKTION, FUNKTIONSFOLGE, ANWENDERPROFIL)
C****      3). NACHSPANN (BEENDEN, AUSWERTEN, PROTOKOLLIEREN)
C****
C**** DER TESTRAHMEN KANN VOM ANWENDER ERWEITERT WERDEN,
C**** MUSS ABER FUER EINE TESTSERIE UNVERAENDERT BLEIBEN.
C****
C*****
C****      == SYSTEM - R O U T I N E N ==
C*****
C****
C****      TIME:   TAGESZEIT (STUNDEN:MINUTEN:SEKUNDEN)
C****      SECNDS: AUSFUEHRUNGSZEIT (SEKUNDEN)
C****
C*****

```

```

C*****
C****      === D A T E N - D E K L A R A T I O N E N ===      ****
C*****
C****
C**** PN = PROGRAMM-IDENTIFIKATIONS-NUMMER (01 ... 99)      ****
C****      FESTE ZUORDNUNG ZU JEDEM TESTPROGRAMM (KONSTANTE)      ****
C**** PI = PROGRAMM-INDEX ZUR STEUERUNG VON PROGRAMMABSCHNITTEN      ****
C****      (PI = -1: PROGRAMM-ABBRUCH)      ****
C****      (PI = 0: OHNE FUNKTIONSTEST)      ****
C****      (PI = 1: MIT FUNKTIONSTEST)      ****
C**** NA = ANZ. AEUSSERE SCHLEIFE (1 ... 32760)      ****
C**** EIN = LOG.-NR.-EINGABE-GERAET (TERMINAL)      ****
C**** AUS = LOG.-NR.-AUSGABE-GERAET (TERMINAL)      ****
C**** PRINT = LOG.-NR.-PROTOKOLL-GERAET (DRUCKER/FERNSCHREIBER)      ****
C**** TA = ZEIT-ANFANG (STD:MIN:SEK)      ****
C**** TE = ZEIT-ENDE (STD:MIN:SEK)      ****
C**** TD = ZEIT-DIFFERENZ (SEK)      ****
C**** STORE = SPEICHER IN DEM DIE ZU DURCHSUCHENDEN BITMUSTER AB-      ****
C****      GELEGT SIND (4096 BYTE)      ****
C****
C**** DA IN FORTRAN I.A. KEIN BYTEZUGRIFF MOEGLICH IST, WERDEN DIE      ****
C**** BYTES IN EINEM INTEGER-ARRAY GEHALTEN. DIE LAENGE N DIESER      ****
C**** FELDES IST VON DER WORTLAENGE DES JEWELIGEN SYSTEMS AB-      ****
C**** HAENGIG: 16 BIT ==> N = 2048 ; 32 BIT ==> N = 1024.      ****
C****
C**** PROZ = ANTEIL DES BITMUSTERS IN STORE IN % (1 ... 50)      ****
C**** ANFANG = ANFANG DES BITFELDES (KONSTANTE: 0 ... 7)      ****
C**** LAENGE = LAENGE DES BITFELDES (KONSTANTE: 1 ... 8)      ****
C**** MUSTER = ZU SUCHENDES BITMUSTER (1 ... 255)      ****
C**** WL = WORTLAENGE DES ZU TESTENDEN RECHNERS IN BIT (16 / 32)      ****
C**** BYTE = ANZAHL DER BYTES PRO WORT DES ZU TESTENDEN RECHNERS      ****
C**** DUMMY = HILFSGROESSE      ****
C****
C*****
C****
      IMPLICIT INTEGER*2 (A-Z)
      INTEGER PN,PI,NA,EIN,AUS,PRINT
      INTEGER TA(4),TE(4)
      INTEGER DUMMY
      INTEGER*2 MUSTER,INHALT,PRUEF,STORE,MASKE
      REAL X
      DIMENSION STORE (2048), MASKE(2)
      REAL TD
      DATA EIN/5/,AUS/6/,PRINT/8/
      DATA PN/6/
      DATA WL/16/,BYTE/2/,N/2048/,ANFANG/2/,LAENGE/5/,VGL/0/
C****
C*****

```

```

C*****
C**** ABSCHNITT 1: === V O R S P A N N ===          ****
C*****
C****
C****          EINLESEN DER PARAMETER              ****
C****
100 WRITE(AUS,101) PN
101 FORMAT(' PN=',I2,' GIB PARAMETER:',
1      /,' =====',
2      /,' PI = -1: PROGRAMM-ABBRUCH',
3      /,'      0: OHNE FUNKTIONSTEST',
4      /,'      1: MIT FUNKTIONSTEST',
5      /,' NA = ANZAHL AEUSSERE SCHLEIFE (1 ... 32760)',
6      /,' MUSTER = ZU SUCHENDES BITMUSTER (1 ... 255)',
7      /,' PROZ = ANTEIL DES BITMUSTERS IN %(1 ... 50)',
8      /,' =====',
9      /,' PI      NA MUSTER PROZ',
A      /,' II IIIII   III   II')
      READ(EIN,102) PI,NA,MUSTER,PROZ
102 FORMAT(I2,X,I5,4X,I3,3X,I2)
C****
C****          PRUEFEN DER PARAMETER              ****
C****
      IF (PI .EQ. -1)      GOTO 90000
      IF (PI .LT. 0)      GOTO 100
      IF (PI .GT. 1)      GOTO 100
      IF (NA .LT. 1)      GOTO 100
      IF (NA .GT. 32760) GOTO 100
      IF (MUSTER .LT. 1) GOTO 100
      IF (MUSTER.GT.255) GOTO 100
      IF (PROZ .LT. 1)    GOTO 100
      IF (PROZ .GT. 50)   GOTO 100
C****
C****          PROTOKOLLIEREN DER PARAMETER        ****
C****
      WRITE(AUS,103) PN,PI,NA,MUSTER,PROZ
      WRITE(PRINT,103) PN,PI,NA,MUSTER,PROZ
103 FORMAT(' PN=',I2,' PI=',I2,' NA=',I5,' MUSTER=',I3,' PROZ=',I2)
C****
C****          BEREITSTELLEN VON MUSTER, MASKE, BLENDE ****
C****
      NHALB = N / 2
      NPLUS = N + 1
      SHIFT = 8 - LAENGE - ANFANG
      MUSTER = MUSTER * 2**SHIFT
      J = MUSTER
      DO 210 I = 1 , BYTE
      MASKE (I) = MUSTER
      IF (I .EQ. BYTE) GOTO 210
      MUSTER = MUSTER * 256
C****          BESSER, WENN ISLL VORHANDEN:        ****
C**** MUSTER = ISLL(MUSTER,8)                      ****
      210 CONTINUE
C*****

```

```

C*****
C****
C**** FUELLEN DES HAUPTSPEICHERBEREICHS MIT BELIEBIGEN BITMUSTERN
C****
      MUSTER = J
      INHALT = 0
      DO 230 I = 1 , N
      K = BYTE - 1
      Z = INHALT
      DO 220 J = 1 , K
      Z = Z * 256
220  CONTINUE
      STORE (I) = Z
C****          BESSER, WENN ISLL VORHANDEN:
C**** STORE(I) = ISLL(INHALT, 8 * (BYTE - 1))
      INHALT = INHALT + 1
      IF (INHALT .EQ. MUSTER) INHALT = INHALT + 1
      IF (INHALT .GE. 256) INHALT = 0
230  CONTINUE
C****
C**** FUELLEN DES HAUPTSPEICHERBEREICHS MIT DEM SUCHMUSTER
C**** MIT VORGEGEBENER (PROZ) DISTANZ (DIST)
C****
      X = 100. / PROZ
      DIST = X
C**** MAXIMALZAHL DER ZU SPEICHERNDEN ZEICHEN:
      X = N * PROZ / 100.
      K = X + 0.5
      J = 0
      DO 240 I = 1 , N , DIST
      STORE (I) = MASKE (BYTE)
      J = J + 1
      IF (J .EQ. K) GOTO 242
240  CONTINUE
242  CONTINUE
C****
C****          START FUNKTIONS-TEST
C****
      WRITE(AUS,190) PN
      WRITE(PRINT,190) PN
190  FORMAT(' PN=',I2,' *** START ***')
C****
C****          ABFRAGE DER STARTZEIT
C****
      CALL TIME (TA)
      TD = SECNDS(0.0)
C****
C*****

```