

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第5621280号
(P5621280)

(45) 発行日 平成26年11月12日 (2014.11.12)

(24) 登録日 平成26年10月3日 (2014.10.3)

(51) Int.Cl.		F I	
G09D	3/12	(2006.01)	G09D 3/12 H
G04G	99/00	(2010.01)	G04G 1/00 311D
G04G	9/00	(2006.01)	G04G 9/00 304C

請求項の数 11 (全 15 頁)

(21) 出願番号	特願2010-48483 (P2010-48483)	(73) 特許権者	000001443
(22) 出願日	平成22年3月5日 (2010.3.5)		カシオ計算機株式会社
(65) 公開番号	特開2011-186009 (P2011-186009A)		東京都渋谷区本町 1 丁目 6 番 2 号
(43) 公開日	平成23年9月22日 (2011.9.22)	(74) 代理人	110001254
審査請求日	平成25年1月28日 (2013.1.28)		特許業務法人光陽国際特許事務所
		(74) 代理人	100090033
			弁理士 荒船 博司
		(74) 代理人	100093045
			弁理士 荒船 良男
		(72) 発明者	梶谷 治男
			東京都羽村市栄町 3 丁目 2 番 1 号 カシオ
			計算機株式会社 羽村技術センター内
		審査官	荒井 隆一

最終頁に続く

(54) 【発明の名称】 カレンダー表示制御装置、カレンダー表示制御方法およびプログラム

(57) 【特許請求の範囲】

【請求項 1】

月日および休日の情報を表示装置に表示させるカレンダー表示制御装置において、
祝日を表わす祝日情報を記憶するデータ記憶手段と、
前記データ記憶手段に記憶された前記祝日情報に基づいて振替休日を算出する第 1 休日
演算手段と、

前記データ記憶手段に記憶された前記祝日情報に基づいて、前日及び翌日が祝日であり
且つ当日が祝日になっていない国民の休日を算出する第 2 休日演算手段と、

前記第 1 休日演算手段および前記第 2 休日演算手段により算出された前記振替休日およ
び前記国民の休日を含んだ休日の情報を前記表示装置に表示させる休日表示制御手段と、

前記データ記憶手段に記憶される前記祝日情報の追加、変更および削除を可能とするデ
ータ変更手段と、

を備え、

前記第 1 休日演算手段と前記第 2 休日演算手段とは、前記データ変更手段により前記祝
日情報の追加、変更又は削除が行われた場合に、追加、変更又は削除の後の前記祝日情報
に基づいて、前記振替休日と前記国民の休日とをそれぞれ算出することを特徴とするカレ
ンダー表示制御装置。

【請求項 2】

前記第 1 休日演算手段は、前記祝日情報と振替休日の有無を判定するための振替休日識
別子の情報とに基づいて前記振替休日を算出し、

10

20

前記第 2 休日演算手段は、前記祝日情報と前記振替休日識別子の情報と国民の休日の有無を判定するための国民休日識別子の情報とに基づいて前記国民の休日を算出することを特徴とする請求項 1 に記載のカレンダー表示制御装置。

【請求項 3】

前記第 1 休日演算手段は、前記第 1 休日演算手段に入力された第 1 入力日が日曜日であるか否かを判断し、日曜日でない場合には前記振替休日識別子がセットされているか否かを判断し、前記振替休日識別子がセットされている場合には前記第 1 入力日を休日と判断し、

前記第 2 休日演算手段は、

前記第 2 休日演算手段に入力された第 2 入力日が日曜日でなく、且つ前記第 2 入力日の前日が前記祝日情報により示される祝日である場合に前記国民休日識別子をセットする国民休日識別子設定部と、

前記第 2 入力日が前記祝日情報により示される祝日であり、且つ前記国民休日識別子がセットされている場合に前記第 2 入力日の前日を休日と判断する前日判定部と、

を備える

ことを特徴とする請求項 2 に記載のカレンダー表示制御装置。

【請求項 4】

前記データ記憶手段には、

ユーザ定義の休日を表わす個人休日情報が前記祝日情報と識別可能に記憶可能であり、

前記データ変更手段は、

前記データ記憶手段に記憶される前記個人休日情報を前記祝日情報と区別して追加、変更および削除可能にされ、

前記休日表示制御手段は、

前記個人休日情報により示される休日を含んだ休日の情報を前記表示装置に表示させることを特徴とする請求項 1 に記載のカレンダー表示制御装置。

【請求項 5】

外部から操作指令を入力する操作手段を備え、

前記データ変更手段は、

前記操作手段を介した外部からの操作指令の入力により前記祝日情報の追加、変更および削除を可能とすることを特徴とする請求項 1 又は請求項 4 に記載のカレンダー表示制御装置。

【請求項 6】

前記データ記憶手段は、不揮発性メモリにより構成されていることを特徴とする請求項 1 又は請求項 4 又は請求項 5 のいずれか一項に記載のカレンダー表示制御装置。

【請求項 7】

月日および休日の情報を表示装置に表示させるカレンダー表示制御装置において、

祝日を表わす祝日情報に基づいて、前日及び翌日が祝日であり且つ当日が祝日になっていない国民の休日を算出する休日演算手段と、

前記休日演算手段により算出された前記国民の休日を含んだ休日の情報を前記表示装置に表示させる休日表示制御手段と、

を備えることを特徴とするカレンダー表示制御装置。

【請求項 8】

祝日を表わす祝日情報を記憶するデータ記憶手段および表示装置をコンピュータが制御して前記表示装置に月日および休日の情報を表示させるカレンダー表示制御方法において、

前記コンピュータが、前記データ記憶手段に記憶された前記祝日情報に基づき振替休日を算出する第 1 休日演算ステップと、

前記コンピュータが、前記データ記憶手段に記憶された前記祝日情報に基づいて前日及び翌日が祝日であり且つ当日が祝日になっていない国民の休日を算出する第 2 休日演算ステップと、

10

20

30

40

50

前記コンピュータが、前記第 1 休日演算ステップおよび前記第 2 休日演算ステップにより算出された前記振替休日および前記国民の休日を含んだ休日の情報を前記表示装置に表示させる休日表示制御ステップと、

前記コンピュータが、操作手段を介した外部からの操作指令の入力に基づき前記データ記憶手段に記憶される前記祝日情報の追加、変更および削除を行うデータ変更ステップと、

を含み、

前記データ変更ステップにより前記祝日情報の追加、変更又は削除が行われた場合に、追加、変更又は削除の後の前記祝日情報に基づいて前記第 1 休日演算ステップと前記第 2 休日演算ステップがそれぞれ実行されることを特徴とするカレンダー表示制御方法。

10

【請求項 9】

前記データ変更ステップは、

前記コンピュータが、前記操作手段を介した外部からの操作指令の入力に基づき、前記データ記憶手段に記憶されているユーザ定義の休日を表わす個人休日情報を、前記祝日情報と区別して追加、変更および削除可能なステップを含み、

前記休日表示制御ステップは、

前記コンピュータが、前記個人休日情報の休日を含んだ休日の情報を前記表示装置に表示させることを特徴とする請求項 8 記載のカレンダー表示制御方法。

【請求項 10】

祝日を表わす祝日情報を記憶するデータ記憶手段と、月日および休日の情報を表示する表示装置とを制御するコンピュータに、

20

前記データ記憶手段に記憶された前記祝日情報に基づいて振替休日を算出する第 1 休日演算機能と、

前記データ記憶手段に記憶された前記祝日情報に基づいて前日及び翌日が祝日であり且つ当日が祝日になっていない国民の休日を算出する第 2 休日演算機能と、

前記第 1 休日演算機能および前記第 2 休日演算機能により算出された前記振替休日および前記国民の休日を含んだ休日の情報を前記表示装置に表示させる休日表示制御機能と、

操作手段を介した外部からの操作指令の入力に基づいて前記データ記憶手段に記憶される前記祝日情報の追加、変化および削除を行うデータ変更機能と、

を実現させ、

30

前記第 1 休日演算機能と前記第 2 休日演算機能とは、前記データ変更機能により前記祝日情報の追加、変更又は削除が行われた場合に、追加、変更又は削除の後の前記祝日情報に基づいて前記振替休日と前記国民の休日とをそれぞれ算出することを特徴とするプログラム。

【請求項 11】

前記データ変更機能は、

操作手段を介した外部からの操作指令の入力に基づき、前記データ記憶手段に記憶されているユーザ定義の休日を表わす個人休日情報を前記祝日情報と区別して追加、変更および削除可能とする機能を含み、

前記休日表示制御機能は、

40

前記個人休日情報の休日を含んだ休日の情報を前記表示装置に表示させることを特徴とする請求項 10 記載のプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

この発明は、月日と休日の情報を表示装置に表示させるカレンダー表示制御装置、カレンダー表示制御方法およびプログラムに関する。

【背景技術】

【0002】

カレンダー表示を行う電子機器において休日の表示を行うものが以前より存在する。

50

【 0 0 0 3 】

また、本発明に関連する先行技術として、特許文献 1 には、祝祭日と日曜日とが重なって生じる振替休日を自動的に算出して表示するカレンダー表示装置について開示されている。また、特許文献 2 には、万年カレンダーのデータと祝祭日の月日データに基づいて、祝祭日が振り替えられたり祝祭日が移動したりする場合に、これらを反映した休日の情報を自動的に算出する装置が開示されている。

【 先行技術文献 】

【 特許文献 】

【 0 0 0 4 】

【 特許文献 1 】 特公昭 6 3 - 6 7 1 5 7 号公報

10

【 特許文献 2 】 特開 2 0 0 0 - 2 0 5 0 0 号公報

【 発明の概要 】

【 発明が解決しようとする課題 】

【 0 0 0 5 】

祝日は永久固定のものでなく変更されることがある。また、日本の休日には、日曜日、祝日、振替休日だけでなく、祝日と祝日に挟まれた平日が休日となる「国民の休日」と呼ばれる休日がある。

【 0 0 0 6 】

この発明の目的は、祝日に変更された場合にもこの変更に対応して正確な休日の表示を行うことのできるカレンダー表示制御装置、カレンダー表示制御方法およびプログラムを提供することにある。

20

【 課題を解決するための手段 】

【 0 0 0 7 】

上記目的を達成するため、請求項 1 記載の発明は、
月日および休日の情報を表示装置に表示させるカレンダー表示制御装置において、
祝日を表わす祝日情報を記憶するデータ記憶手段と、
前記データ記憶手段に記憶された前記祝日情報に基づいて振替休日を算出する第 1 休日演算手段と、

前記データ記憶手段に記憶された前記祝日情報に基づいて、前日及び翌日が祝日であり且つ当日が祝日になっていない国民の休日を算出する第 2 休日演算手段と、

30

前記第 1 休日演算手段および前記第 2 休日演算手段により算出された前記振替休日および前記国民の休日を含んだ休日の情報を前記表示装置に表示させる休日表示制御手段と、

前記データ記憶手段に記憶される前記祝日情報の追加、変更および削除を可能とするデータ変更手段と、

を備え、

前記第 1 休日演算手段と前記第 2 休日演算手段とは、前記データ変更手段により前記祝日情報の追加、変更又は削除が行われた場合に、追加、変更又は削除の後の前記祝日情報に基づいて、前記振替休日と前記国民の休日とをそれぞれ算出することを特徴としている。

【 発明の効果 】

40

【 0 0 1 5 】

本発明に従うと、祝日に変更が生じた場合には、データ記憶手段に記憶される祝日情報の追加、変更または削除によって対応することができる。また、振替休日と国民の休日は祝日情報から演算により求めているので、祝日の変更に伴ってこれら振替休日と国民の休日とを共に含んだ正確な休日の情報が求められるという効果がある。

【 図面の簡単な説明 】

【 0 0 1 6 】

【 図 1 】 本発明の実施形態のカレンダー表示装置の全体構成を示すブロック図である。

【 図 2 】 カレンダー表示装置の表示画面を示す正面図である。

【 図 3 】 不揮発性メモリに記憶される祝日データと個人休日データの一例を示すデータチ

50

ャートである。

【図４】ＣＰＵにより実行される休日データ変更処理の制御手順を示すフローチャートである。

【図５】ＣＰＵにより実行される休日表示処理の制御手順を示すフローチャートの第１部である。

【図６】同、休日表示処理のフローチャートの第２部である。

【図７】同、休日表示処理のフローチャートの第３部である。

【発明を実施するための形態】

【００１７】

以下、本発明の実施の形態を図面に基づいて説明する。

10

【００１８】

図１は、本発明の実施形態のカレンダー表示装置の全体構成を示すブロック図、図２は、このカレンダー表示装置の表示画面を示す正面図である。

【００１９】

この実施形態のカレンダー表示装置１は、現在日時と現在日を含む複数日分（例えば１週間分）の日付、曜日および休日の情報を表示する装置であり、図１と図２に示すように、表示画面２０Ａに上記表示が行われる表示部２０と、表示制御を行うＣＰＵ（中央演算処理装置）１０と、ＣＰＵ１０が実行する制御プログラムおよび制御データが格納されたＲＯＭ（Read Only Memory）１１と、ＣＰＵ１０に作業用のメモリ空間を提供するＲＡＭ（Random Access Memory）１２と、書き換え可能な制御データを格納するデータ記憶手段としての不揮発性メモリ１３と、現在日時を計数する計時回路１４と、複数の操作ボタンを有し外部から操作指令を入力する操作部（操作手段）１５等を備えている。上記構成のうち、表示部２０を除いた各構成によりカレンダー表示制御装置が、ＣＰＵ１０、ＲＯＭ１１、ＲＡＭ１２によりコンピュータが、それぞれ構成される。

20

【００２０】

表示部２０の表示画面２０Ａには、時刻等が表示される第１表示領域２１と、カレンダー表示が行われる第２表示領域２２とが設けられている。第１表示領域２１には、通常時において現在時刻が表示される一方、操作部１５を介して設定データの入力が行われる際にはデータ入力用の表示が行われる。

【００２１】

30

第２表示領域２２には、左の範囲に現在の年月日が表示される年月日表示領域２２１が設けられ、中央から右の範囲に一週間分のカレンダー表示が行われるカレンダー表示領域２２２が設けられている。

【００２２】

カレンダー表示領域２２２には、その下段に一週間分の日付表示Ｄ１がなされ、中段に一週間分の曜日表示Ｄ２が固定的になされ、上段に休日であることを表わす休日マークＤ３が表示され、日付表示Ｄ１の上部に現在の日付を指し示す指示マークＤ４が表示される。日付表示Ｄ１は、「１～３１」の１桁又は２桁の日付数字を７組一列に並べて構成され、各々の日付数字の表示はＣＰＵ１０の制御によって変更可能にされている。曜日表示Ｄ２は、各曜日を表わす曜日文字が日付表示Ｄ１の各日付数字にそれぞれ対応する位置に、７個一列に並べられて構成される。指示マークＤ４は、日付表示Ｄ１の７組の日付数字の上部にそれぞれ表示のオンとオフの切り換えが可能にされ、休日マークＤ３は、曜日表示Ｄ２の７組の曜日文字の上部にそれぞれ表示のオンとオフの切り換えが可能にされている。

40

【００２３】

通常表示モードの際、このカレンダー表示領域２２２には、一週間が経過するごとに一週間分の日付数字と休日マークＤ３の表示のオン・オフが更新されて、現在日を含む一週間分のカレンダー表示がなされるようになっていく。また、日付が変わるごとに指示マークＤ４の表示オンされる位置がシフトして、常に現在日が指示された状態にされるようになっていく。

50

【 0 0 2 4 】

図 3 には、不揮発性メモリ 1 3 に構築される祝日データテーブル 1 3 1 と個人休日データテーブル 1 3 4 の一例を表わすデータチャートを示す。

【 0 0 2 5 】

不揮発性メモリ 1 3 には、祝日を表わす祝日情報が複数登録された祝日データテーブル 1 3 1 と、ユーザ定義の休日を表わす個人休日情報が複数登録可能にされた個人休日データテーブル 1 3 4 とが設けられている。

【 0 0 2 6 】

祝日データテーブル 1 3 1 には、複数年分の春分の日と秋分の日のデータが登録された春分秋分の日データテーブル 1 3 2 と、毎年 of 祝日データが登録された年次データテーブル 1 3 3 とが含まれる。年次データテーブル 1 3 3 には、データ登録されていないブランク領域が複数用意され、新たに祝日が追加されたときに、祝日情報を追加登録することが可能になっている。

10

【 0 0 2 7 】

春分秋分の日データテーブル 1 3 2 には、例えば、年月日が指定された祝日情報が登録され、年次データテーブル 1 3 3 には、例えば、月日のみが指定された祝日情報や、月と第何週の第何曜日という形式で指定された祝日情報等が登録されている。

【 0 0 2 8 】

個人休日データテーブル 1 3 4 には、ユーザ定義の休日が種々の形式で指定された休日情報が登録可能になっている。例えば、曜日のみ指定の休日情報、年月日やその範囲が指定される休日情報、月日やその範囲が指定される休日情報、月と第何週の第何曜日という形式で指定される休日情報などが登録可能になっている。個人休日データテーブル 1 3 4 には、データ登録されていないブランク領域が複数用意され、新たな休日情報を追加登録することが可能になっている。

20

【 0 0 2 9 】

R O M 1 1 には、制御プログラムとして、計時回路 1 4 の計時データに基づいて第 1 表示領域 2 1 の時刻表示や第 2 表示領域 2 2 のカレンダー表示の更新を行う日時表示処理プログラム、操作部 1 5 を介した操作指令の入力に基づいて動作モードを通常の日時表示モードから休日データ変更モードやその他のモードへ変更するモード切替制御プログラム、操作部 1 5 からの入力によって祝日データテーブル 1 3 1 と個人休日データテーブル 1 3 4 のデータの追加、変更および削除が可能な休日データ変更処理プログラムなどが格納されている。また、上記の各プログラムの中から呼び出されてカレンダー表示領域 2 2 2 の表示を更新する 1 週間の休日表示処理プログラムも格納されている。

30

【 0 0 3 0 】

これらの各プログラムは、R O M 1 1 に格納するほか、例えば、データ読取装置を介して C P U 1 0 が読み取り可能な、光ディスク等の可搬型記憶媒体、フラッシュメモリ等の不揮発性メモリに格納しておくことが可能である。また、このようなプログラムがキャリアウェーブ（搬送波）を媒体として通信回線を介してカレンダー表示装置 1 にダウンロードされる形態を適用することもできる。

【 0 0 3 1 】

次に、上記構成のカレンダー表示装置 1 において実行される休日データ変更処理（データ変更手段）および 1 週間の休日表示処理について説明する。

40

【 0 0 3 2 】

図 4 には、C P U 1 0 により実行される休日データ変更処理のフローチャートを示す。

【 0 0 3 3 】

この休日データ変更処理は、操作部 1 5 を介して動作モードが休日データ変更モードに変更されたことで開始される。休日データ変更処理が開始されると、まず、C P U 1 0 は、表示部 2 0 へ表示データを出力して第 1 表示領域 2 1 の表示をデータ入力用の表示に切り替える（ステップ S 1 ）。

【 0 0 3 4 】

50

次に、CPU 10は、操作部 15を介した操作入力を受け付けて、祝日データテーブル 131の祝日情報の追加、変更および削除を行うのか、個人休日データテーブル 134の個人休日情報の追加、変更および削除を行うのか、ユーザに選択入力を行わせる（ステップ S2）。具体的には、ユーザが操作部 15を操作して祝日情報か個人休日情報の何れかを選択し、その後、操作部 15を操作して選択完了の操作を行うことで、上記の選択入力

【0035】

データテーブルの選択入力が行われたら、続いて、CPU 10は、操作部 15を介した操作入力を受け付けて、祝日データテーブル 131又は個人休日データテーブル 134のうち選択されたデータテーブルにおけるデータ登録先のメモリ番号（データの登録領域）の選択入力を行わせる（ステップ S3）。そして、選択されたメモリ番号にデータが登録されていれば、そのデータを表示部 20の第 1表示領域 21へ表示させる（ステップ S4）。なお、このステップ S3、S4の処理を、メモリ番号の選択完了の操作が行われるまでループ処理により繰り返し行われるようにしても良い。それにより、ユーザはメモリ番号の選択中に選択途中のメモリ番号に何のデータが登録されているのか確認しながら、データの追加、変更および削除の対象とするメモリ番号を選択することが可能となる。

【0036】

メモリ番号の選択入力およびそのデータ表示が行われたら、続いて、CPU 10は、操作部 15を介した操作入力を受け付けて、ユーザにデータ入力形式の選択入力を行わせる（ステップ S5）。すなわち、月日指定、年月日指定、これらの範囲指定、曜日指定、月と第何週の第何曜日の指定など、選択されているメモリ番号に登録可能なデータ形式のうち、何れかを選択させる。

【0037】

そして、データ入力形式の選択入力が行われたら、続いて、CPU 10は、操作部 15を介した操作入力を受け付けて、上記選択されたデータ形式により指定されるデータの入力をユーザにより行わせる（ステップ S6）。具体的には、ユーザが操作部 15を操作して、年月日、曜日、第何週などの指定データを 1つずつ順送りしていき、所望の値に送られたときに操作部 15を介して確定の操作を行う。このような操作を選択されたデータ形式に含まれる指定データの個数分（例えば、年月日の指定なら“年”と“月”と“日”の 3個分）行うことでデータの入力が遂行される。また、このステップ S6のデータ入力処理において、指定データを空白にしたまま入力完了の操作を行うことで、空白データの入力となって、元に登録されていたデータを削除することもできる。

【0038】

次に、CPU 10は、ステップ S2で選択された祝日データテーブル 131又は個人休日データテーブル 134の何れかのデータテーブルにおいて、ステップ S4で選択されたメモリ番号のデータ登録領域へ、ステップ S6の入力データを上書きして登録させる。なお、ステップ S6で空白データが入力されている場合には、選択されたメモリ番号に元々登録されていたデータが削除されることとなる。

【0039】

入力データを登録したら、次いで、CPU 10は、新規に登録のあったデータテーブル（祝日データテーブル 131又は個人休日データテーブル 134）について複数の登録データを所定条件に従った順番に並べかえるソート処理を行う（ステップ S8）。例えば、祝日データテーブル 131の年次データテーブル 133については、月日の若い順に登録データが並べ替えられ、且つ、空白領域が下位に来るようにされる。また、個人休日データテーブル 134については、曜日指定の登録データ、月日指定の登録データ、年月日指定の登録データの順で且つ各指定データの値が若い順に登録データが並べ替えられ、さらに、空白領域が下位に来るようにされる。

【0040】

ソート処理が完了したら、CPU 10は、表示部 20へ表示データを出力して第 1表示

10

20

30

40

50

領域 2 1 の表示をデータ入力用の表示から通常の時刻表示へ戻す（ステップ S 9）。さらに、第 2 表示領域 2 2 のカレンダー表示についても表示の更新が行われるように表示更新要求を発効する（ステップ S 10）。この表示更新要求の発効により、続いて実行される日時表示処理において、週や日付の変更がなくても第 2 表示領域 2 2 のカレンダー表示に対して日付や休日の抽出処理から行われて、カレンダー表示が更新される。そして、このカレンダー表示の更新により、上記ステップ S 2 ～ S 8 で更新された祝日情報や個人休日情報が反映された休日マーク D 3 の表示のオン・オフ制御がなされるようになっている。その後、この休日データ変更処理を終了する。

【 0 0 4 1 】

図 5 ～ 図 7 には、CPU 10 により実行される 1 週間の休日表示処理のフローチャートを示す。

10

【 0 0 4 2 】

この 1 週間の休日表示処理は、1 週間の経過（土曜日から日曜日への変更時）或いは日付の変更時、並びに、上記の表示更新要求の発効により、カレンダー表示を更新する際に併せて実行されるものであり、カレンダー表示中の一週間分の休日マーク D 3 のオン・オフを決定する処理である。

【 0 0 4 3 】

この 1 週間の休日表示処理が開始されると、まず、CPU 10 は、計時回路 1 4 の計時データの年月日の情報に基づいて、現在日を含む日曜日から土曜日までの一週間の日付を抽出するとともに、表示部 2 0 へ表示データを出力して、これらの日付値を反映した日付表示 D 1 を行わせる（ステップ S 2 1）。

20

【 0 0 4 4 】

次に、CPU 10 は、振替休日の有無を判定するための振替フラグ F 1、国民の休日の有無を判定するための国民休日フラグ F K 1、F K 2 を共にクリアする初期化处理（ステップ S 2 2）と、Xレジスタにステップ S 2 1 で抽出された一週間の日付のうち最初（日曜日）の日付データをセットし、Yレジスタに祝日データテーブル 1 3 1 の最初のメモリ番号に登録されている祝日情報をセットする（ステップ S 2 3）。Xレジスタは、この後、日付表示 D 1 に出力される一週間分の日付値が順にセットされる RAM 1 2 内のレジスタであり、Yレジスタは、この後、祝日データテーブル 1 3 1 の祝日情報が順にセットされる RAM 1 2 内のレジスタである。XレジスタとYレジスタにデータがセットされたら、次に進む。

30

【 0 0 4 5 】

次に進むと、CPU 10 は、Xレジスタのデータ値とYレジスタのデータ値の比較を行って、Xレジスタにセットされた日付がYレジスタにセットされた祝日と重なるか判別する（ステップ S 2 4）。その結果、重なっていれば（一致）、Xレジスタの日付に対応した休日マーク D 3 を表示オンにする（ステップ S 2 5：休日表示制御手段）。一方、不一致であれば、Yレジスタに祝日データテーブル 1 3 1 の全ての祝日情報がセットされて比較終了済みか確認し（ステップ S 2 6）、未だであればYレジスタに次の祝日情報をセットして（ステップ S 2 7）、再び、ステップ S 2 4 に戻る。

【 0 0 4 6 】

40

すなわち、ステップ S 2 4、S 2 6、S 2 7 のループ処理が繰り返し行われることで、Xレジスタの日付が祝日データテーブル 1 3 1 に登録されている複数の祝日の何れかと重なっているか否かが判別されて、祝日と重なっていればステップ S 2 5 で休日マーク D 3 が表示オンされ、何れの祝日でもなければステップ S 2 6 の判別処理で“YES”側へ分岐するようになっている。

【 0 0 4 7 】

その結果、Xレジスタの日付が何れの祝日でもなくループ処理を抜けたら、CPU 10 は、この日付が日曜日か判別し（ステップ S 2 8）、日曜日でなければ振替休日を示す振替フラグ F 1 がセットされているか否かを確認する（ステップ S 2 9）。そして、日曜であるか、或いは、振替フラグ F 1 がセットされていれば、振替フラグ F 1 をクリアした上

50

で（ステップS30）、Xレジスタの日付に対応した休日マークD3を表示オンにする（ステップS31：休日表示制御手段）。

【0048】

一方、日曜でもなく、振替フラグF1のセットもなければ、CPU10は、国民休日フラグFK1のセットが有るか判別し（ステップS32）、有れば国民休日フラグFK2をセットするとともに国民休日フラグFK1をクリアする（ステップS33）。一方、国民休日フラグFK1のセットが無ければ、国民休日フラグFK2をクリアして（ステップS34）、次に進む。ここで、国民休日フラグFK1とは、前日が祝日である場合にセットされるフラグであり、国民休日フラグFK2とは、前日が祝日で且つ当日が祝日でも日曜日でもないときにセットされるフラグである。

10

【0049】

次に進むと、CPU10は、Xレジスタの日付に対応した休日マークD3を表示オフにして（ステップS35：休日表示制御手段）、図6のステップS42へジャンプする。そして、Xレジスタの日付が土曜日（日付表示D1の最終日）であるかを確認し（ステップS42）、土曜日でなければ、Xレジスタの日付を「+1日」更新し、Yレジスタに祝日データテーブル131の最初のメモリ番号に登録されている祝日情報をセットする（ステップS43）。それから、再び、ステップS24に戻る。

【0050】

つまり、上記のステップS24～S42、S43のループ処理が繰り返されることで、Xレジスタの日付が日付表示D1の先頭の日付から最後の日付まで1日ずつ変更されて、休日か否かの判定処理が行われるようになっている。

20

【0051】

一方、ステップS24の判別処理で一致（祝日）と判別されて、ステップS25でXレジスタの日付に対応する休日マークを表示オンしたら、次いで、Xレジスタの日付が日曜日か判別し（ステップS36）、日曜日であれば次の日が振替休日となるので振替フラグF1をセットして（ステップS41）、ステップS42に移行する。

【0052】

一方、日曜日でなければ、国民休日フラグFK2がセットされているか判別し（ステップS32）、セットされていれば、Xレジスタの日付を当日とすれば、前々日が祝日、前日が平日、当日が祝日であり、前日が「国民の休日」であると判定されるので、Xレジスタの日付の前日に対応する休日マークを表示オンにする（ステップS40：休日表示制御手段）。そして、ステップS42に移行する。

30

【0053】

一方、日曜日でもなく、国民休日フラグFK2のセットもなければ、振替フラグF1がセットされているか判別し（ステップS38）、セットされていれば、祝日が続いていて、さらに次の日を振替休日とするために、そのままステップS42へ移行する。また、振替フラグF1がセットされていなければ、次の日が「国民の休日」となりえるので国民休日フラグFK1をセットして（ステップS39）、それからステップS42へ移行する。

【0054】

ステップS42へ移行したら、上述したように、Xレジスタの日付が土曜日であるかを確認（ステップS42）と、土曜日でなければ、Xレジスタの値を更新するとともにのYレジスタの値を初期化してステップS24に戻る。

40

【0055】

そして、Xレジスタの日付が一週間分更新されて、ステップS42の判別処理で土曜日であると判別された場合には、一週間の最後の土曜日が「国民の休日」となっていないか判定する処理（ステップS44～S49）に進む。

【0056】

すなわち、まず、CPU10は、国民休日フラグFK2がセットされているか判別する（ステップS44）。このフラグFK2がセットされていなければ、一週間の最後の土曜日が「国民の休日」になりえないので、そのまま、図7のステップS50へ移行する。

50

【 0 0 5 7 】

一方、フラグ F K 2 がセットされていれば、Xレジスタの日付を「+ 1 日」更新し、Yレジスタに祝日データテーブル 1 3 1 の最初のメモリ番号に登録されている祝日情報をセットする（ステップ S 4 5）。ここで、Xレジスタにセットされる日付は、一週間の日付表示 D 1 に表れていない、その次の日曜日の日付となる。

【 0 0 5 8 】

次いで、C P U 1 0 は、ステップ S 4 6 , S 4 7 , S 4 9 のループ処理を繰り返して、Xレジスタの日付が祝日データテーブル 1 3 1 に登録されている何れかの祝日に該当しないか判別する処理を行う。上述したステップ S 2 4 , S 2 6 , S 2 7 のループ処理と同様の処理である。

10

【 0 0 5 9 】

このループ処理中、Xレジスタの日付が祝日に該当していれば、ステップ S 4 6 の判別処理で一致の側へ移行する。この場合、前日の土曜日が「国民の休日」に該当するので、土曜日の休日マーク D 3 を表示オンにして（ステップ S 4 8 : 休日表示制御手段）、図 7 のステップ S 5 0 へ移行する。

【 0 0 6 0 】

一方、上記のループ処理で、Xレジスタの日付が祝日に該当していなければ、ステップ S 4 7 の判別処理で全ての祝日情報と比較終了と判別され、この場合、前日の土曜日は「国民の休日」に該当しないことになるので、そのまま、図 7 のステップ S 5 0 へ移行する。

20

【 0 0 6 1 】

以上のステップ S 2 1 ~ S 4 9（図 5 , 図 6）の処理によって、カレンダー表示される 1 週間の各日付が、祝日、日曜日、振替休日、国民の休日の何れかに該当するか算出されて、算出結果に応じた休日マーク D 3 の表示が遂行されるようになっている。上記のステップ S 2 4 , S 2 6 , S 3 6 , S 4 1 , S 2 8 ~ S 3 1 等、振替フラグ F 1 に関わる処理により第 1 休日演算手段が構成され、ステップ S 3 2 ~ S 3 4 , S 3 6 ~ S 4 0 , S 4 4 ~ S 4 9 等、国民休日フラグ F K 1 , F K 2 に関わる処理により第 2 休日演算手段が構成される。

【 0 0 6 2 】

続いて C P U 1 0 により実行されるステップ S 5 0 ~ S 5 7（図 7）の処理は、カレンダー表示される 1 週間の各日付が、ユーザ定義の個人休日に該当するか否か確認して、個人休日に該当する場合に、対応する休日マーク D 3 の表示をオンさせる処理である。個人休日については、祝日と異なって振替休日や国民の休日を発生させることがないので、ステップ S 2 1 ~ S 4 9 の祝日に関する処理と別途分けて実行されるようになっている。

30

【 0 0 6 3 】

ステップ S 5 0 に移行すると、C P U 1 0 は、まず、Xレジスタにステップ S 1 で抽出されている一週間の日付のうち最初（日曜日）の日付のデータをセットし、Yレジスタに個人休日データテーブルの最初のメモリ番号に登録されている個人休日情報をセットする（ステップ S 5 0）。

【 0 0 6 4 】

続いて、C P U 1 0 は、Xレジスタの日付がすでに休日に設定されているか判別し（ステップ S 5 1）、未だ休日に設定されていなければ、Xレジスタの日付がYレジスタの個人休日情報の条件に合致するか否かを判別する（ステップ S 5 2）。

40

【 0 0 6 5 】

その結果、Xレジスタの日付が休日設定されてなく、Yレジスタの個人休日情報の条件に合致したと判別されたら、Xレジスタの日付に対応した休日マーク D 3 を表示オンにする（ステップ S 5 3 : 休日表示制御手段）。一方、Xレジスタの日付がすでに休日設定されているか、Yレジスタの個人休日情報の条件に合致していないと判別されたら、Yレジスタに個人休日データテーブル 1 3 4 の全ての個人休日情報をセットして比較終了済みを確認し（ステップ S 5 4）、未だであればYレジスタに次の個人休日情報をセットして（

50

ステップ S 5 5)、再び、ステップ S 5 1 に戻る。

【 0 0 6 6 】

すなわち、上記のステップ S 5 1、S 5 2、S 5 4、S 5 5 のループ処理が繰り返し行われることで、Xレジスタの日付がユーザ定義の休日に設定されているか否かが判別されて、ユーザ定義の休日に設定されていれば、この日付に対応した休日マーク D 3 が表示オンされるようになっている。

【 0 0 6 7 】

そして、ステップ S 5 2 の判別処理で個人休日情報の条件と合致したと判別されるか、或いは、ステップ S 5 4 の判別処理で全ての個人休日情報との比較が終了したと判別されるかして上記のループ処理を抜けると、C P U 1 0 は、Xレジスタの日付が土曜日（日付表示 D 1 の最終日）であるかを確認し（ステップ S 5 6）、土曜日でなければ、Xレジスタの日付を「+ 1 日」更新し、Yレジスタに個人休日データテーブル 1 3 4 の最初のメモリ番号に登録されている個人休日情報をセットする（ステップ S 5 7）。そして、再び、ステップ S 5 1 に戻る。

【 0 0 6 8 】

つまり、上記のステップ S 5 1 ~ S 5 6、S 5 7 のループ処理が繰り返されることで、Xレジスタの日付が日付表示 D 1 の先頭の日付から最終の日付まで 1 日ずつ変更されて、各日付がユーザ定義の休日か否かの判定が行われて、ユーザ定義の休日であればこの日付に対応した休日マーク D 3 が表示オンされるようになっている。

【 0 0 6 9 】

上記の 1 週間の休日表示処理（図 5 ~ 図 7）が、週や日にちの切り替り時点、または、祝日情報或いは個人休日情報の変更操作後に行われることで、カレンダー表示領域 2 2 2 に当日を含む一週間に日付表示 D 1 とそれに対応した正確な休日情報を表わす休日マーク D 3 とが常に表示されるようになっている。

【 0 0 7 0 】

以上のように、この実施形態のカレンダー表示装置 1 によれば、上述の休日データ変更処理（図 4）によって、祝日データテーブル 1 3 1 の祝日情報の追加、変更および削除を行うことが可能になっている。さらに、祝日情報の追加、変更または削除がなされた場合には、カレンダー表示領域 2 2 2 の表示の更新がなされて、新たな祝日情報に基づく休日マーク D 3 の表示が行われるようになっている。このとき、振替休日と国民の休日についても新たな祝日情報に基づいて算出されて休日の有無が判定されるので、振替休日と国民の休日とを含んだ正確な休日情報を表示することが可能になっている。

【 0 0 7 1 】

また、この実施形態のカレンダー表示装置 1 によれば、ユーザ定義の休日を示す個人休日情報が登録される個人休日データテーブル 1 3 4 が設けられ、この個人休日情報に基づく休日の表示も行われる。また、個人休日情報と祝日情報とは区別されて、個人休日情報に基づいて振替休日や国民の休日が誤って算出されてしまうということがないので、ユーザ定義の休日も含めて正確な休日情報の表示を行うことができる。

【 0 0 7 2 】

また、休日データ変更処理は、操作部 1 5 を介した外部操作により、祝日情報や個人休日情報の変更を行うことが可能なものなので、エンドユーザによって祝日やユーザ定義の休日の設定変更を容易に行わせることができる。また、祝日データテーブル 1 3 1 や個人休日データテーブル 1 3 4 は、不揮発性メモリ 1 3 に設けられているので、登録データの変更が可能であり、且つ電源電圧が遮断された場合でも登録データを失わない。

【 0 0 7 3 】

なお、本発明は、上記実施の形態に限られるものではなく、様々な変更が可能である。例えば、上記実施形態では、表示部 2 0 を有し実際にカレンダー表示を行う装置を例示したが、表示装置を別構成とし、外部の表示装置に表示データや表示信号を出力してカレンダー表示を行わせる表示制御装置に本発明を適用しても良い。また、上記実施形態では、カレンダー表示として、一週間分の日付および休日が表示される形式を例示したが、一ヶ

10

20

30

40

50

月分の日付および休日が表示される形式としても良い。また、当日 1 日の日付および休日が表示される日めくりカレンダーの形式としたり、当日を中心或いは先頭等とした一週間分や所定日数分の表示とするなど、表示形式は種々に変更可能である。

【 0 0 7 4 】

また、上記実施形態では、祝日情報と個人休日情報とを区別するために、別々のデータテーブルを用意して各データテーブルに別個に登録する構成を例示したが、例えば、各休日情報に祝日情報と個人休日情報とを区別する識別データを付加することで、同一のデータテーブルに両者を登録しても、これら両者を区別可能とする構成を適用することもできる。

【 0 0 7 5 】

10

その他、カレンダー表示の画面構成、振替休日と国民の休日との算出方法、祝日情報や個人休日情報の変更方法など、この実施形態に示した細部構成および細部方法は、発明の趣旨を逸脱しない範囲で適宜変更可能である。

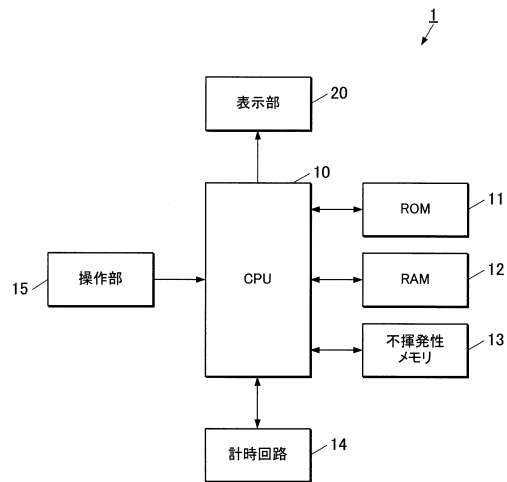
【符号の説明】

【 0 0 7 6 】

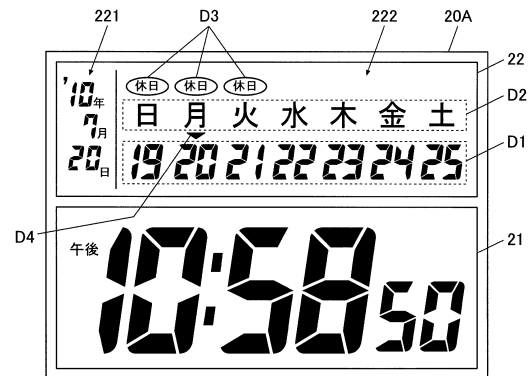
- 1 カレンダー表示装置
- 1 0 C P U
- 1 1 R O M
- 1 2 R A M
- 1 3 不揮発性メモリ
- 1 4 計時回路
- 1 5 操作部
- 2 0 表示部
- 1 3 1 祝日データテーブル
- 1 3 4 個人休日データテーブル
- 2 2 2 カレンダー表示領域
- D 1 日付表示
- D 2 曜日表示
- D 3 休日マーク

20

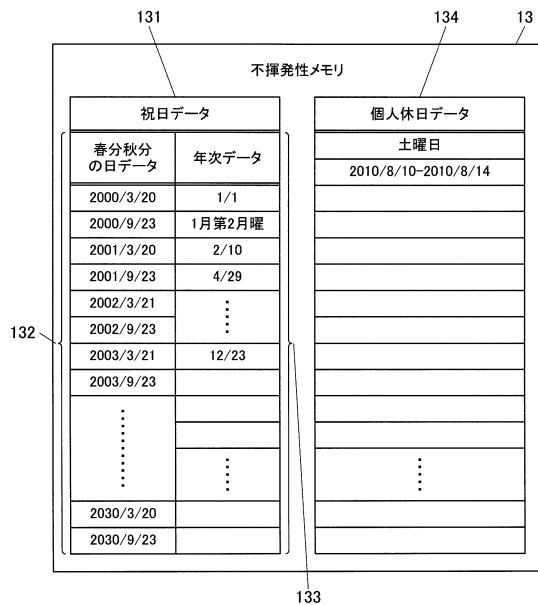
【図 1】



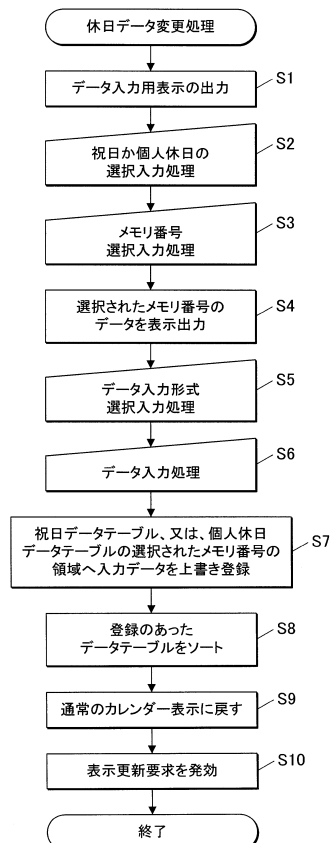
【図 2】



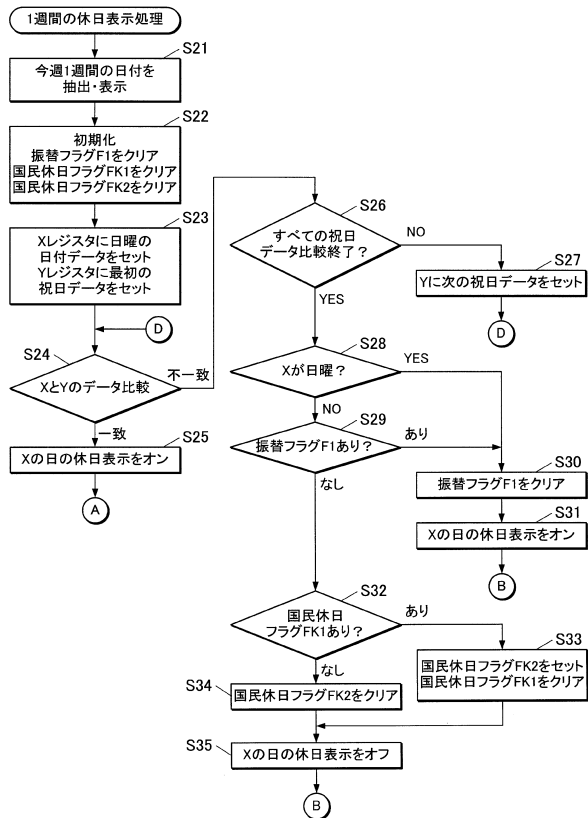
【図 3】



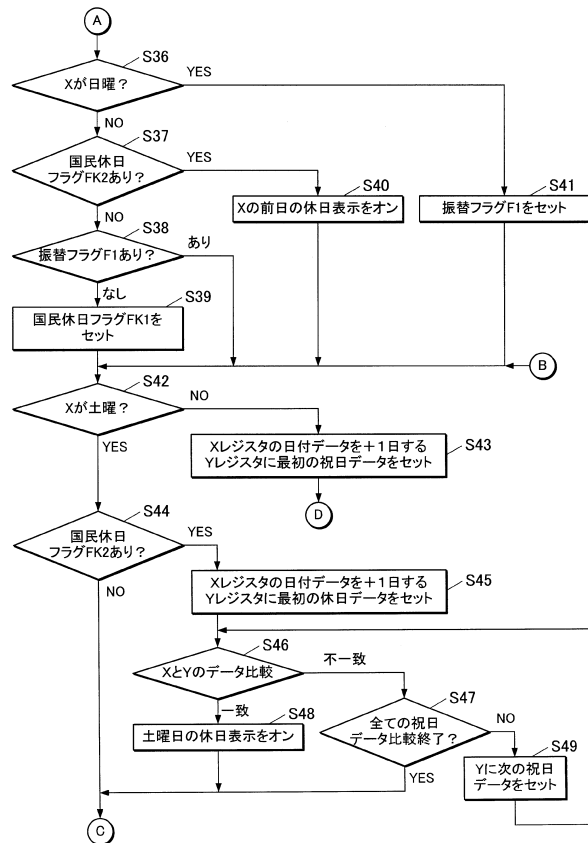
【図 4】



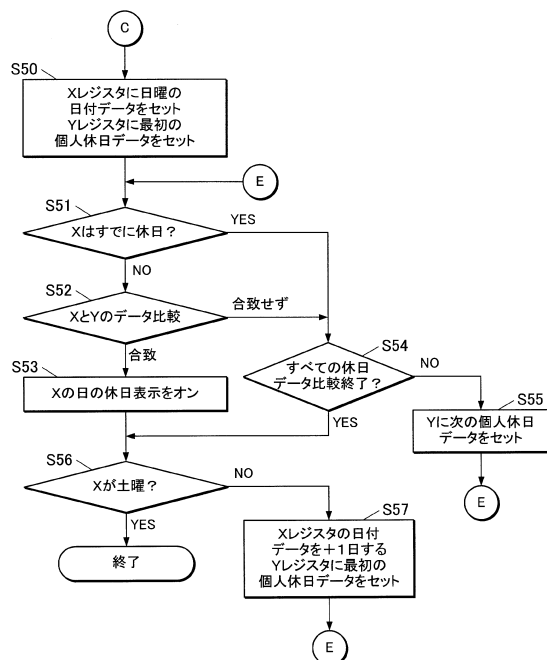
【図 5】



【図 6】



【図 7】



フロントページの続き

(56)参考文献 登録実用新案第3079330(JP,U)
特開2000-20500(JP,A)

(58)調査した分野(Int.Cl., DB名)

G09D 3/00 - 3/12

G04G 9/00 - 9/12

G04G 99/00