Dokumentation & Exegese zum LC80-Projekt „numeral-inversion“

Aufgabenstellung:

*Eine maximal vierstellige Zahl ist dezimal einzugeben und im Display von rechts beginnend dezimal einzutragen. Ist die Zahl negativ, so ist sie maximal dreistellig - jede Stellenzahl bis 4 ist möglich - das MINUS wird durch einen Bindestrich optisch abgebildet! Alle Tasten außer den Ziffern 0 bis 9, sowie PLUS und MINUS sind gesperrt - Quittungstaste ist "EX".*

Versionshinweise:

Die Version v2 stellt eine Weiterentwicklung sowie Konkretisierung des Projektes dar. Ausgangsversion, welche auch lauffähig aber nicht exakt der Aufgabenstellung entspricht, ist die Version v1. Bei dieser Version wird die Inversion nicht von links, sondern von rechts beginnend eingetragen. Zur Bewertung soll die Version v2 herzugezogen werden.

Bedienung:

Die mit ASide aus der ASM-Datei erstellte HEX-Datei wird in die LC80-Simulation importiert. Anschließend mittels der „ADR“-Taste zu der Startadresse 02000H navigieren. Auf entsprechender RAM-Zelle die „EX“-Taste betätigen. Das Programm startet. Nun können 4 Zahlen mit den Tasten 0 bis 9 eingegeben werden. Das Minuszeichen ist mittels der „A“-Taste einzugeben. Die reale Eingabe wird in Echtzeit in dem Display angezeigt. Nach der Eingabe verharrt die Anzeige für eine kurze Zeit in der Darstellung der Eingabesequenz. Anschließend wird die Eingabe gemäß der Aufgabenstellung rotiert sowie von rechts beginnend im Display eingetragen.

Rechtfertigung:

Alle algorithmischen Schritte werden in der ASM-Datei mittels Kommentare beschrieben.

In der Aufgabenstellung werden Eingaben negativer Dezimalzahlen gefordert. Dafür sind außer den Tasten 0 bis 9 nur die Plus- sowie Minustaste erlaubt. Daraus ließe sich folgern, dass die Eingabe des negativen Vorzeichens mittels der Minustaste realisiert wird. Die konzipierte Eingaberoutine mittels DAK1 liefert jedoch bei der Betätigung der Minustaste den gleichen HEX-Wert wie bei der Betätigung der „1“-Taste zurück. Die „1“-Taste ist jedoch obligat zur Eingabe der Dezimalzahlen. Schlussendlich kann die Eingabe des negativen Vorzeichens aufgrund der fehlenden internen Unterscheidbarkeit zweier genannter Tasten nicht realisiert werden.

Als logische Konsequenz ergibt sich, dass eine mittelbare Lösung gefunden werden musste. Da es sich um Dezimalzahlen handelt, welche eingegeben werden sollen, werden nur die Tasten 0 bis 9 benötigt. Die Tasten A bis F liegen ohne eine primäre Notwendigkeit vor. Deshalb wird die Codierung der Taste A von dem Prozessor beziehungsweise dem Assembler als Minuszeichen gewertet oder interpretiert.

Ein generelles Programmierparadigma fordert eine Konditionalverzweigung zur Umwandlung der hexadezimalen 10 in ein Minuszeichen. Diese Möglichkeit ist in Assembler nicht gegeben. Jedoch sind relative Sprünge sowie Zero-Flags als Pseudoevent für einen relativen Sprung möglich. Der Hexadezimalwert A beträgt umgewandelt in einen Dezimalwert 10. Wenn von 10 der eingegebene einstellige Dezimalwert subtrahiert wird, so erhält man nie 0. Wenn aber die Taste A betätigt wird, erhält man bei der Subtraktion von 10 minus hex(a) das Ergebnis 0. Das Zero-Flag wird hochgehalten (entsprechendes Bit des Flagregisters F wird 1). Somit kann man in zwei relative Sprünge unterscheiden: Einmal die Umwandlung der Eingabe mittels ONESEG bei „NZ“ und zweitens (sonst) die direkte Übergabe eines Minuszeichens.

Die beschriebene Aufweichung der Aufgabenstellung fand unter Absprache statt.