The Ni-Wumpf Replacement CPU Manual

Revision F

Overview

Design

- Considerations
- Implementation
- BIT Test

Installation Caveats

- Mounting
- Position
- WARNING!
- Configuration

Diagnostic Procedures

- Bookkeeping
- Tests
- Options

Troubleshooting

- Fault Isolation

Warranty

Schematics

- **Ni-Wumpf Replacement CPU Layout Diagram**
- **Schematic Diagram (Sheet 1 of 3)**
- **Schematic Diagram (Sheet 2 of 3)**
- **Schematic Diagram (Sheet 3 of 3)**
- **Parts List**

Overview

A lot of manuals start off by saying "...Congratulations on your purchase of the Iludium Q36 explosive space modulator...", and run off at the mouth from there. But if you are reading this one, you have found yourselves in the highly rarefied area of space reserved for technical hackers trying to repair a set of 16 games from Gottlieb back in the late 70's that didn't even use a CPU in the modern sense of the word. People like us do not read manuals, (*especially* the Overview), we install the equipment, tinker fruitlessly for hours, find out that it baffles the mind absolutely, and return to consult the manual as a last ditch effort reserved only for those times when all else has failed already. The only way that the overview would ever be consulted is in the sense of "*light*" reading material after the module has proven to work wonderfully, or lies in a mangled heap of IC's and cap's, next to the application form soon to be addressed to the Better Business Bureau.

So, given the optimistic outlook that the board is already installed and working beautifully, and that the reader is merely interested in some background material on what could possibly have gone into the creation of such a useful piece of equipment, let me elucidate somewhat further into this area...

It doesn't take a Masters in Computer Science to realize that there are a **lot** of different games in the pinball universe, and that the set of System 1 games from Gottlieb in 1977 - 1979, comprise only a very small niche in this universe. Unlike most though, I am one of those oh-so-fortunate few to have purchased my very first pinball game, "**Sinbad**", from this group. As such, I had always had a special interest in getting this game running just right. Originally, it came to me from a vendor, who quite fairly, described it as having a "memory problem" that he couldn't fix. It was nearly a year later that I knew enough about the architecture of the CPU to replace the 5101 chip, and finally see the game remember it's high score table. By that time, I had acquired three System 1 games, and it was getting increasingly more important that I know enough about these CPU's to keep the games running. Until finally, the "**Charlie's Angels**" that I had just purchased, arrived with a very seriously dead CPU. To top it off, the spare CPU's I had managed to scavenge for parts had no way of running either, they were just that; parts. About 6 months after this, I was able to determine, in my infinite wisdom, that the part that was actually bad was an extremely odd-looking "spider" chip labeled 1752, and not much more. Research is not my strongest suite, so it took me a bit of time to discover that this was, in fact, a custom VLSI chip, and not available from any of the crack electronic components shoppes in my area. Not one to be daunted by this, I made my first call to the factory; Premier (who apparently was formed from Gottlieb). Did they have this chip? Oh yes, three left; $75 apiece. I have to admit, I was inches from my wallet and the purchase of these extremely rare pieces of silicon. But I deferred, and went back into the "*lab*" to see what I could do to part out these spare boards and get one working. This was 1990. In 1991, I no longer had any spare boards with a functional 1752 chip in it, and I knew enough to be able to identify a faulty 1753 chip in a big sort of hurry as well. It was plain then that unless something happened to rectify this situation, that there was not a big future for my dear-to-the-heart "**Sinbad**". Sooner of later, that chip would fail, and the game was done; forever; no resuscitation. By this time, I knew the full architecture of the CPU, and with the help of Steve Young and his wealth of background material, I knew exactly what those "spider" chips were, and why Rockwell designed a 4-bit distributed architecture CPU back then before the Z80 silicon, was widely available. And I also knew that Rockwell was perfectly willing to press these chips again for us, if we would front them fifty grand for a minimum lot order. Again I was close to reaching for my wallet, though how much good it would have done me is dubious when you consider my account might have had about six hundred dollars in it at the time. My eyes were opened to the end of an era in 1992, when I realized that no one was going to save these games and that the System 1 Gottlieb games were rapidly being consigned to the dump by wild-eyed operators with a nervous twitch in their grins, and a song in their hearts, madly whipping System 1 components and playfields against the far wall of the dumpster. Off and on I thought about what I could do to redesign this CPU, but hey, let's face it, I was never that hot a hardware techy, and the concept of a PIA was vague at best, with no idea of why you might actually need one. Had I not managed to pervert a very good friend of mine from high-school, to the hobby of collecting video games and pinball machines, (like I was), we'd all be waiting for someone, somewhere, who knew what they were doing, to develop this CPU sooner or later. Now, this was no easy task I can assure you, considering this friend of mine lived with his wife in a trailer! (How good a friend can you be, to convince someone that forcing a bunch of pinball games into a living space just big enough for Iguanas, is that next, very important, phase of life that everyone just has to go through...) As it turns out, this is one of those guys that finds it mildly interesting to force the functionality of a Cray supercomputer into one of those "gate array" thingies, and go out for pizza after it's done. Just right. He convinced me that designing such a CPU, was relatively sweatless; and his wife assured me that he would prove it. And so it began...

The one thing that was immediately raised to our attention when test-marketing the idea of a replacement CPU, was the interface cable between the CPU board and the driver board. On 85% of the System 1 games that I have restored, this connector has had to be repaired with new pins, and the contacts on the CPU board burnished to get a good surface. Most thought we should integrate the CPU and driver board, thus eliminating this connector entirely. This combination would have been the ultimate replacement for the Gottlieb System 1 games, and was kicked around up and down the hallway for quite a while before the final resolution. Cost. All of those lamp driver transistors and Solenoid transistors would have raised the cost for a replacement CPU to a point that I felt would have been beyond the threshold of endurance for someone who was making that important decision of: buy a new CPU, or trash the game entirely. Since the sole reason for making this board was to prevent the loss of these games from the face of the planet, we couldn't risk that decision going the wrong way because it cost too much.

The only way we could resolve the issue of this Achilles' heel of a connector in the System 1 games was to attack what was at the heart of the problem in the first place; the System 1 NiCad battery. After no little analysis, you can quickly come to the conclusion that this is the direct cause of these games failing from the very start; corrosion from the gaseous leaks of this battery.

The postulate goes something like this; leakage of this battery slowly destroys the lower half of the CPU, as well as the cable connector to the driver board. After time, the connection to the driver board becomes so affected by this corrosion that contact is lost. Contact is very important to the driver board, because it tells the solenoid drive transistors to keep the solenoid off! Without it, the input to the transistors floats to any particular state that it likes, and it *likes* to be on! So now, the game is running, the contact is lost and some solenoid in the playfield is *very* energized. After some relatively short time, the transistor gets tired of pumping those amps through a nearly dead short to ground and fries up tight. Apparently these are transistors that tend to look skyward when fusing up, and make a close relationship with the +43 volts that the solenoids run off of, because this +43 volts then becomes immediately available to the CPU. Now while the connection to the driver board is good and corroded to the +5 volt level, it is always ready to go that extra mile for the +43 volt impetus. So, the end result is to see the full solenoid voltage introduced to an unwilling driver circuit at the old 7417 chips. Now, unlike the drive transistors, this transistor logic had a less likely chance of fusing the drive output to the input line, but it did happen often enough to pass that voltage level, directly or indirectly, to the next chip in line, the 1753 chip, and thus the story ends; no more 1753's. All because of one lousy battery, and the fact that no one was around to turn the game off once it was noticed that the smoke from that toasty solenoid was obscuring game play.

So to remove the root of all evil once and for all, and make it no longer necessary to repair this connector (more than once) that nobody wants, the design of the new CPU incorporates EEPROM technology to eliminate the battery entirely. The design of the new CPU should handle all of the known problems of the System 1 games, and make it possible to once again reach a comfortable level of reliability with these games (not to mention the world being safe for democracy).

Design

- Considerations...

The design of the replacement System 1 board was premised first and foremost on the capability for long-term component replacement. If you read the overview preceding this, you would have an idea as to just why this is true. The second most important concept was to utilize as simple a design as possible for ease of trouble-shooting in the future; and also for reliability. The basic tenet behind this one was the loose rule of MTBF logistics; the fewer the components, the longer the Mean Time Between Failure becomes. Realistically though, these two rules wound up clashing in the design, and we had to make heated, often violent discussions termed "compromises" between the two. I wanted lot's of nifty little 74-series TTL logic chips to make the board up with, and Pete wanted to shove four or five of these chips into one PAL and roll it out like that. I argued that it would be easier to trace the logic when isolating component failure, and Pete didn't bother arguing at all. Instead, he made me work with the prototype he had breadboarded. After the second prototype, he no longer had to defend his position, because I didn't have a leg to stand on, and all I had left to do then, was justify in my mind how I would present this decision to use PAL's to you, the reader. Philosophically, I don't like PAL's, but practically, I have found them *very* useful, and they **do** present a path to a higher MTBF. The religion is this; the PAL is a black box of signals going in and signals coming out. If you are trying to troubleshoot a component failure, you can stop right there at that PAL, and without knowing what is going on inside, you can't usually progress past that chip if it's condition is unknown. Empirically, if the signal coming out of that chip is stuck at high-Z, you can pretty much guess it's bad, but if it is locked high or low, it **may** or may **not** be bad... Well, it turns out that the component count on the CPU is so low as to minimize this issue, so after the final board layout was complete, I wasn't too upset with the decision.

- Implementation...

The processor is based on the ubiquitous Z80 CPU which is widely available and relatively inexpensive; this is closely tied into Intel's 8279, (a PIA) traditionally used for a keyboard controller (*inside* the keyboard of most <u>all</u> PC's, so this too is pretty easy to obtain). The 8279 also handles the display memory and refresh, so after I tell it to display the number "7" in column 5 of player 2's display, it does the rest. (easy, eh?), and what is more, it is doing all of the keyboard (in this case switch matrix) scanning and encoding. The memory of the 8279 is continuously being dumped to the displays and timed via the 74154 demultiplexer that creates a strobe line for each of the 16 digit positions on all of the four displays, along with the status display. The character information is arranged, like the System 1 board, into bank A (the upper four bits) and bank B (the lower four bits) of the 8-line output bus. Each of these banks, forms it's character via the first two PAL's for the eight-segment displays. Now, here is where the PAL decision was a very <u>big</u> bonus. We could have used 7448-type seven-segment decoders, but had a lot more fun with the PAL's, programming in this same functionality, and expanding on it, as you will see in the diagnostic test section! On the switch section of operation, the same strobes that strobe the display digits, are used to strobe the playfield switch matrix. The 74156 chip decodes three of the four strobe lines and produces the demultiplexed 8 strobe lines (of which only 5 are used). The pull-ups on the return lines function to tie the open-collector 156 chip to +5, as well as speed up the recovery time of the switch strobe signal after switch closure/grounding. And the capacitor bank aids in filtering signal spikes coming in on the return lines. Which about sums up the input and output sections of the board to the 8279.

The RAM memory is a widely available 6116 type 2k RAM, and is address selected, along with the other components of the board, by yet another PAL chip. The program ROM is stored in a 32-kByte EPROM and serves as the storage area for both the system ROM, as well as the game specific code. The last bit of "RAM" on board, is the EEPROM memory used to save the option settings, the thresholds, bookkeeping, and the high score to date. This is a serial EEPROM, thus the pin count is kept to a minimum. Since this EEPROM has some finite limit to the amount of times it can be successfully written to, most of the bookkeeping is updated at the end of a game (except credits). Keep this in mind if you are really getting the high score to end all high scores and you suddenly think that plugging in the air-conditioner on the same circuit as the game is a really neat idea.

Output to the driver board for the lamps, is performed by a 7416 to buffer the data lines, as well as 2 PALs to select one of the nine lamp latches on the driver board, from the address lines requested. The solenoid pulse is latched in a 74374 (which is also selected via a PAL), and uses the same 7416's to drive the solenoid lines. In revision 'B' of the CPU, the above lamp data strobe lines and the solenoid output lines were augmented with zener diode protection, to shunt voltage feedback from faulty driver boards.

Lastly, the CPU is clocked by a 3.276MHz crystal divided by a 74HCT74 and is synched in still another PAL to generate the system reset to the CPU.

- BIT Test

On power-up, the system will reset and start it's first section of code, which is to blink the LED on the board once. The next self-test is to verify that the ROM meets it's specified checksum, followed by a test of the available RAM memory, a test of the EEPROM with <u>it's</u> check-sum, and lastly, an inquiry to the 8279 for it's status. Should any of these tests fail, (save that for the EEPROM), the board will not go into operation. A failure of any of these major components is shown by the number of blinks the LED will blink <u>after</u> the initial blink; one for a ROM failure; two for a RAM failure, three for a detected EEPROM error, and four for an 8279 failure. At the final stage, the CPU then reads four of the 8 DIP switches to determine which game is to be loaded for operation. If for some reason, the game selected is *not* supported by the EPROM installed, the system will blink five times, but continue operation, it is incapable of proper game play, but able to perform bookkeeping and diagnostics. Finally, the game number selected will be displayed in the number-of-balls display, and the game will then be ready to play.

Be aware that the full BIT diagnostic procedure lasts around six seconds after the first blink before the system indicates success, so have some patience on power-up. (though, since this is not any different than the original CPU, it should come as no surprise) The EEPROM diagnostic will fail when the contents of the EEPROM does not match that of the checksum it keeps. This condition can occur during game play when the system is powered-down unexpectedly. After a checksum failure is detected, however, the system will re-calculate the new checksum to match its contents. On the next self-test the EEPROM will then pass its diagnostics if the component itself is not faulty. However, if two consecutive self-tests fail on the EEPROM, it indicates a true component failure.

- Installation Caveats

The System 1 replacement CPU was designed to be a plug-in replacement for the original System 1 CPU, but as can be seen, the size of the board is about one-half that of the original. This brings up issues in several areas; mounting, cabling, and position. The first, and easiest topic to address, is that of cabling. The installer will immediately notice that the connectors on the CPU (figure 1) are located at different positions than those on the original CPU. Because each connector is slotted exclusively for the cable going onto it, it is felt to be impossible to get the wrong cable on the wrong connector. But for those in need of clarification...

The CPU is oriented so that the component side faces out. The long side of the board with two connectors faces down, leaving the left side with two connectors also. On the upper left of the board goes the power cable from the power supply (J1); lower left is the connector for the playfield switch matrix (J7). On the bottom side of the board to the left, is the connector to the front door switches (J6) next to the connector for the driver board (J5), to the right. On the right side of the board is half of the two connectors going to the displays. To install the display cables, connect the lower of the two cables (J3), to the connector on the left, fold the cable over the top of the board, and connect the upper cable (J2), onto the top-most connection..

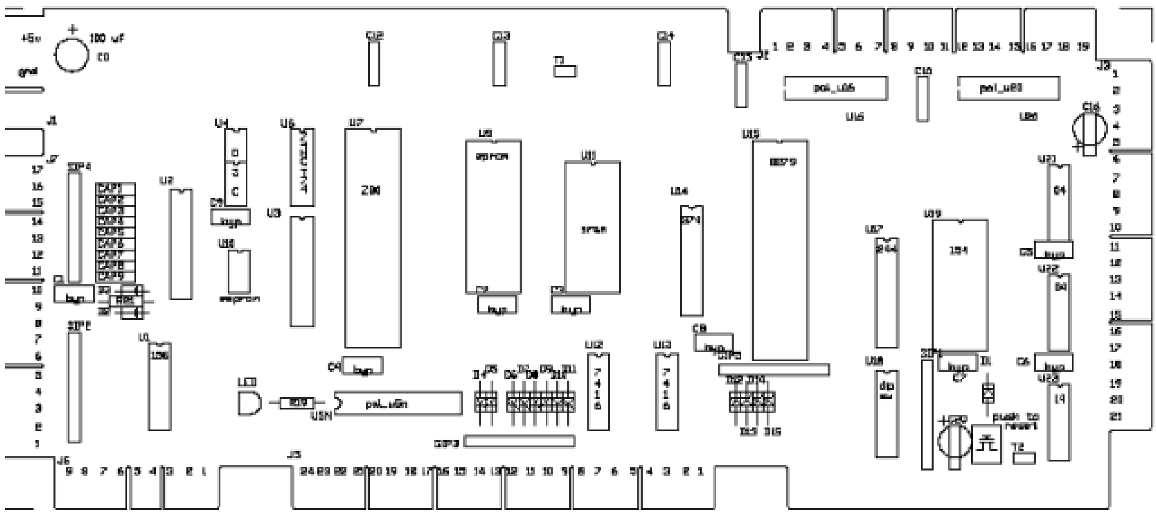


Figure 1 - Ni-Wumpf replacement CPU showing connector designation.

- Mounting...

There are several options one must explore to satisfactorily mount the new System 1 board in the head, and these are brought about by the mounting posts used to hold both boards in place. The upper two registration holes on the new board will line up nicely with the middle two mounting posts used on the old board. However, if you line up the new board to be mounted there, you will quickly notice that the lower two posts do <u>not</u> line up with the lower two holes on the new board. There are basically three options to <u>get</u> the posts to line up with the new holes.

- First: push the new board onto the center two posts, and just let it hang there without worrying about the lower two mounting posts! This is a perfectly fine option for those people that do not wish to mess too much with the head of the system or simply want to be able to re-install the old board at any given time. It is, however, not too great for a game about to be shipped anywhere.
- Second (preferred): remove the lower of the three mounting bars screwed into the back of the head that hold the mounting posts (three wood screws secure it in place). Remove the nylon posts from their original position, and re-install these posts in the holes already in the bar that line up with the mounting holes on the new CPU. Re-attach the mounting bar, and the CPU will be very firmly mounted in place, when pressed onto the posts.
- Third: cut the lower right nylon mount off at the shoulder so that it does not interfere with the placement of the new CPU. Install two new nylon mounts in the holes that line up with the lower holes on the new CPU, and press the CPU in place. Contact your local distributor for availability of these nylon posts..

- Position...

The new CPU mounts in such a fashion that the position of the old cables no longer resembles where they used to be (except for the inter-board connector to the driver board). After removing and installing these cables countlessly while troubleshooting the board, it was determined that this really has no effect on the cables whatsoever, and that no additional strain should be placed on the cables. The logical placement of the cables is the same, left to right, as the old System 1 board, however, the physical placement is a bit different, with the power connector at the upper left of the board; the playfield connector below that; then the front door connector is attached to the bottom of the board on the left, while the inter-board connector is to the right of this. On the right-hand side of the board is the lower display connection, and on the top of the board to the right is the upper display connector. The only curious cable position is this of the display connector which must now bend at a 90 degree turn between the two connectors to attach correctly.

-WARNING-
-WARNING-
-WARNING-

BEFORE the game is powered up with the new CPU there are several guidelines that must be followed to prevent the board from suffering damage.

First, and foremost; if the interboard connector between the CPU and the driver board (J5) has suffered corrosion damage, the pins on this cable should be replaced. Corroded pins will be brittle, and prone to imminent failure, while the lack of spring tension on a corroded pin will also make for poor contact. Granted, the CPU has circuitry that is a bit more immune to the type of failure that bad pin connectors can cause, it is still very possible to short out this board with a bad driver board, or a bad cable.

Second; a bad display will destroy the new CPU just as readily as it did the old CPU. However, on the new CPU, the display driver chips are custom PAL's that we produce, and are not as easily obtained as 7448's are, (though *those* are hard enough to come by in themselves). These PALs are basically only available from ***Ni-Wumpf***, and your distributor, which makes getting them that much longer a process. Please be sure that all of the displays to be used are checked out before you power up the board. I know that this can be a difficult procedure, especially because to check out an unknown display, means possibly frying the test CPU if it is bad. Fortunately, there are not that many bad displays that fall into this category. To err on the side of safety is the best approach, and this means not to use a display whose functionality is questionable. If you have displays that fall into this "unknown" category, please contact your distributor to find the availability of a custom display checker from ***Ni-Wumpf***.

Third, the connectors; J6 and J7 are just as likely to be corroded as J5, and the pins, just as brittle. While these connectors cannot seriously damage the CPU because of their condition, it is imperative that they be checked and repaired to attain proper game operation.

Fourth, no connector should be removed from the CPU while it is powered up! This can damage the CPU, especially the display cables. Always power-down the CPU before removing any connectors.

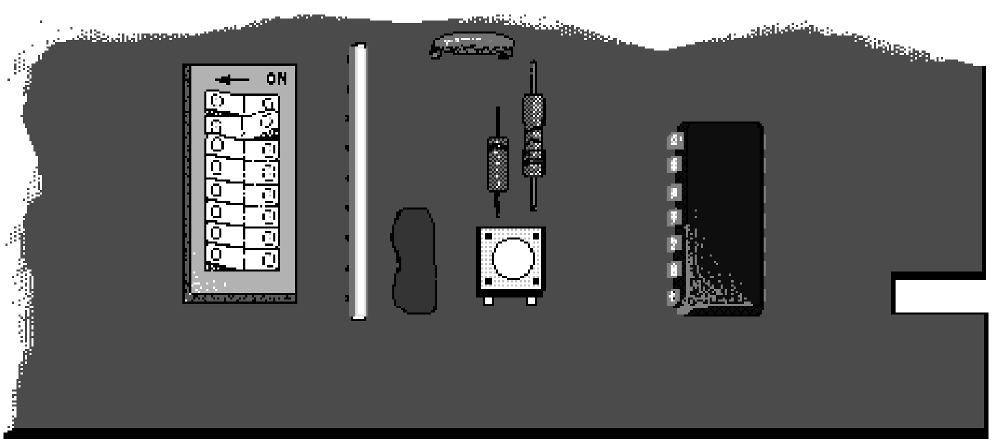


Figure 2 - Location of the DIP switch (Sinbad selected).

- Configuration...

The CPU is configured to operate one of the sixteen System 1 games via the DIP switch at the lower right.

The upper four switches are used to set, in binary, the game number for the game you want to use. Because the EPROM will contain game code for **all** of the System 1 games, the CPU will read this switch on power-up and select which game code to load in. Please be aware that *not all of the revisions of the* EPROM support all of the sixteen games. If a game is selected that is not supported by that revision of the EPROM, the LED on the CPU will flash five times, and refuse to load any game code. Table 1 shows the games made by Gottlieb and their corresponding game number to dial into the DIP switch. Should your EPROM not support the game you have selected, please contact your distributor, or the factory to find out how you can get your code updated.

Game	ID	DIP1	DIP2	DIP3	DIP4
Cleopatra	1	on	off	off	off
Sinbad	2	off	on	off	off
Joker Poker	3	on	on	off	off
Dragon	4	off	off	on	off
Solar Ride	5	on	off	on	off
Count Down	6	off	on	on	off
Close Encounters	7	on	on	on	off
Charlie's Angels	8	off	off	off	on
Pinball Pool	9	on	off	off	on
Totem	10	off	on	off	on
Incredible Hulk	11	on	on	off	on
Genie	12	off	off	on	on
Buck Rogers	13	on	off	on	on
Torch	14	off	on	on	on
Roller Disco	15	on	on	on	on
Asteroid Annie	0	off	off	off	off

Table 1 - Game identification via DIP switch settings

Figure 2 shows the position of the DIP switch on the board and the locations of switch 1 through 8. Switch number 1 is at the topmost location, and switch number 8 is located at the bottom. The on/off position of the switch should be clearly marked on the switch itself.

The code supplied with the Ni-Wumpf CPU is not always compatible with the operation of the original CPU. In some cases, the changes were intentional, others are a result of modern circuitry, and still others are the result of too many Margaritas, low blood sugar, and a callous disregard for proper coding technique when operating a pinball machine. One of these intentional coding changes may be overridden by setting the appropriate DIP switch. This is the case of the "special" routine on many of the games.

In *all* games where the prerequisite amount of unique combination of challenging shots have been achieved, the player is awarded with the chance to score a "special". On some of the tables, however, the game may then proceed to reset the lights, rollovers, and drop targets necessary to collect that special again. Normally, however, while the "special" lamp may light yet another time, the game will *not* award a second special. One of the changes made to the code was to correct this lighting "flaw", and enable the player to achieve as many "specials" on one ball as he is capable of scoring. However, when these games are used on location, this may result in earnings loss. Therefore, to override this software "feature", DIP switch number 6 may be set "on" to enforce, an alternate "special" award for use on location.

In this mode, upon completion of a second "special" within a single ball playing time, the award is changed from the default of an extra game or ball, to one of adding 20,000 points to the players score. The setting of this value can be modified in the "dATA" section of diagnostics.

Diagnostic Rituals...

The diagnostic routine for the CPU board is organized under three separate sub-menus from the test routine. When you first press the test switch inside the game's front door, you should see the displays blank, and the player 1 display show the word **dATA** in rough alphanumerics. (I don't want to hear any hassles about how "well" the 8-segment displays in the game display letters, so just keep that off the complaints category!) The **dATA** menu is that used for gathering and resetting bookkeeping statistics for the game. Pressing the test switch again will offer the operator the chance to enter the **hArdWr** sub-menu, used for testing the operational condition of the various parts to the game. The last sub-menu available, is the **oPTIon** menu, and like the original CPU, this gives you the chance to set and modify different playing/bookkeeping options in the game. When the test switch is pressed yet again, the game will give the operator the chance to exit the test mode; **donE**, and resume game play. To enter into any of the sub-menus displayed in the player 1 display, *press the replay button* on the front door. This will select the option shown, whether it is a menu, or the **donE** command.

When a sub-menu has been selected in the player 1 display, the player 2 display will begin to display the fields available for that sub-menu, while the player 3 display will show the value of that field. To advance the field displayed in the player 2 display, depress the test button. To increment/cycle the value displayed, repeatedly press the replay button. To zero this value press the test button and the replay button simultaneously. (In this case "simultaneously" means to be pressed as closely spaced in time as can be expected. The easiest method to accomplish this is to press and hold the replay button, then depress the test button, and then quickly release both.) Selecting the **donE** field in a sub-menu, will return the user to the main menu, clearing the player 2 and 3 displays. To advance onto the next field, after modification, depress the test button once again.

- dATA

In the bookkeeping sub-menu, the following fields are available for interrogation:

Field	**Description**
crEdS	number of credits remaining
STArTS	number of games started
coInSL	number of coins through left chute
coInSr	number of coins through right chute
rEPLAS	number of replays awarded
chEATS	number of slam tilts detected
nUTS	number of free balls awarded
TILTS	number of normal tilts detected
Th onE	first replay threshold
Th TWo	second replay threshold
TThrEE	third replay threshold
hI onE	high score to date
SPEcIL	points awarded for alternate special

Table 2 - Bookkeeping fields and their meaning

This is the only sub-menu where depressing the replay button, and holding it closed will be detected by the CPU and will auto-increment the following fields; **Th onE**, **Th TWo**, **TThrEE**, and **hI onE**. The rest of the fields can only be zeroed when changed. Bookkeeping information is completely reset when the selection is made to restore factory settings (see below).

- hARdWR

In the hardware testing sub-menu, the CPU can test the operation of the displays, the lamps, the coils, and the switches. While exercising any of the tests below (except for the switch test), any switch closure on the playfield will interrupt the running test, and terminate it.

dISPS - Selecting the display test will begin the first half of the display test, which is to count from 0 to 9 repetitively in every display digit for all of the displays. If you depress the replay switch again during this test you will toggle the test from this half of the test, to the second half, which is to show a marching digit across each of the display digits counting from 0 to 9 repetitively.

LITES - Selecting the lamp test will continuously pulse the lamps on and off for around three minutes before timing out.

coILS - Selecting the coil test will cycle through all of the system coils available, including the special coil assignments used in such games as Count Down and Close Encounters. While the coils are pulsing, the corresponding coil number will be displayed in the player three display. Table 3 will detail the coil assignment numbers. Please note that not all games use coils 6, 7, and 8. Consult your game specific manual for the game being tested to determine the coil assignment number to the playfield solenoid. This exercise will repeat also for about three minutes before timing out.

SWITch - In this test, the CPU scans the playfield for any newly detected switch closures and reports the switch assignment in the player 3 display. One second after the last switch is detected, the display will return to show a "0" in the display, indicating that the CPU is scanning for newly detected closures. If two or more switches are depressed at once, the CPU will display them, but you will only be able to see the last switch indication in the display. Be aware that this test cannot determine if any switches have been stuck closed since before the test began, as it can only detect newly activated switches.

InIT E - Selecting this function restores factory defaults to the EEPROM. The CPU will flash the displays when it is done resetting the audits, high-scores, and features.

Coil Assignment	Coil
1	outhole
2	knocker
3	10’s chime
4	100’s chime
5	1,000’s chime
6	<game specific>
7	<game specific>
8	<game specific>
9	lamp 17 aux. (if used)
10	lamp 18 aux. (if used)
11	game over relay
12	tilt relay

Table 3 - Coil assignments

- oPTIonS

If you consult the original game manual for the System 1 game, you will find that all of the parameters that were selectable via the DIP switches on that board are also selectable via this menu, with a few added fields at the end. It was felt that some of the System 1 games would benefit from an add-a-ball ability, and so that option was added. It was also determined that the entire diagnostic set-up to the games had gotten just complex enough to justify an option to reset all of the EEPROM settings to the factory default.

Field	Description	Possible Values
coUnTL	left coin counter multiplier.	(see table 5 for matrix)
coUnTr	right coin counter multiplier.	(see table 5 for matrix)
ToP	Maximum credits allowed.	0 - 15; 0 enables free play
LoTTo	Match feature	0 - disabled; 1 - enabled
ThrEE	three balls/game	0 - five balls; 1 - three
SPEcIL	special award	0 - extra ball; 1 - replay
TILT	tilt penalty	0 - game over; 1 - loose ball
ShoW c	show credits	0 - disables; 1 - enabled
c TunE	play credit tune	0 - disabled; 1 - enabled
S TunE	play startup tune	0 - disabled; 1 - enabled
chuTE	chute control	0 - allow separate coin chute control 1 - set coin chute 2 control to be that of chute 1
ShoWhI	display high game to date	0 - don't display high score during idle 1 - enable display
AWArd	award for beating HSTD	0 - no award for beating high score to date 1 - three replays awarded
Add	add-a-ball	0 - disabled; 1 - enabled

Table 4 - Options settings and their descriptions

Setting	Description
0	one coin/credit
1	one coin / 2 credits
2	one coin / 3 credits
3	one coin / 4 credits
4	one coin / 5 credits
5	one coin / 6 credits
6	one coin / 7 credits
7	one coin / 8 credits
8	one coin / 9 credits
9	two coins/credit
10	two coins / 2 credits[5]
11	two coins / 3 credits[5]
12	two coins / 4 credits[5]
13	two coins / 5 credits[5]
14	two coins / 3 credits
15	three coins/credit

Table 5 - Coin counter multiplier table.

Table 5 - Coin counter multiplier table.(cont.)

Troubleshooting

- Fault Isolation

Isolating what section of the CPU is broken in the case of failure is necessary before the IC that may be responsible can be replaced. The table below is far from complete, as can be expected, so should some symptom exist that does not fall into any of these categories, you may contact the support number for further assistance.

The CPU should normally blink *once* during power-up to indicate that it is operational. Following this single flash, it should play a start-up tune after about 5 seconds. The CPU performs a number of start-up tests during this time to determine the operation of the peripheral controller (8279), the non-volatile RAM (93C46), the RAM (6118), and the program ROM (27256). If any of these components, or their supportive circuitry, do not respond as expected, the CPU flashes the LED an additional number of diagnostic blinks to attempt to indicate the problem.

Symptom	Possible Cause
LED lit continuously on power-up	Verify that the following default signals can be seen at the points indicated; a clock signal at pin 6 of U7, a reset signal at pin 26 of U7 (when the reset switch is depressed), and of course, power is up at +5. Should any of these prerequisites be lacking, you will have to track back this problem prior to any further board troubleshooting. (Refer to schematics).
	Probe pin 16 (pin 12 of a 16v8 configuration) of U2, EPROM select; if no activity is seen at the pin, probe pin 18(pin 14 of a 16v8) of U2, ~MREQ; if no activity there, hit the reset switch; if still no pulse at pin 18[10], replace U7. Probe pins 1, and 19, of U17, these should remain high, if not skip to issue on “LED flashes three times”, below. Probe pin 16 of U7, this should remain high (unless the Z80 is servicing an interrupt from the 8279), if low, check the operation of the inverter gate in U23 (pins 9 and 8), if this is OK, replace U15, otherwise replace U23. If, after the checks above, pin 16 *is* selecting U9 and the LED is lit; replace U9.
	If replacement of these parts does not fix the problem check the data and address lines for shorts or grounds. A stuck address/data line can normally be caused by U7, U17, U9, or U11, in order of probability. However, in the case where a known solenoid problem has occurred, D0, D1, D2, and D3, at U12, U13, or U14, can be hung due to chip failure there. If problem still persists, replace U7.
LED flashes continuously on power-up	Check that pin 17 of U2 actively selects U11. If so, Replace U11, else replace U2.
LED flashes once after power-up.	Check that pin 16 of U2 actively selects U9. If so,

	Replace U9, else replace U2.
LED flashes twice after power-up	Probe pin 18 of U11, if no activity is present during power-up self-test, replace U2, else replace U11.
LED flashes three times after power-up	Probe pins 18 and 19 of U3 during power-up self-test cycle, check for ~2.4 microsecond pulses indicating both chip enable, and R/W to U10. If incorrect, replace U3[14], else check pin 1 of U17 for similar pulses enabling read of the U10, if incorrect replace U3[14], if correct, check pin 2 of U17 for data from U10, if correct replace U17, if incorrect replace EEPROM (U10).
LED flashes four times after power-up.	Remove connectors J6 and J7 from CPU. Try self-test again. If this solves the problem, there may be a problem with the switch matrix being stuck. If not, power-down game completely. Re-apply power, and check self-test again. If still bad, probe pin 22 of U2 for activity, if nothing replace U2, otherwise, replace U15.
LED flashes five times after power-up	DIP switch configuration is incorrectly set.
Lamps missing strobe column (1 of 9)	Verify operation of output at pins 14 - 22 of U5 during lamp test, if one or more is not cycling, replace U5, else the problem may lie in the driver board.
Lamp missing row (1 of 4)	Check appropriate inverter in U12 or U13, if operation is correct check resistance values of R9 - R12. If correct, problem may be in driver board, otherwise, replace U12/U13.
Display segment always high or missing in displays of bank A	Remove connection to each display and verify problem exists on each display, if so, replace U16, if not, one display may be affecting the entire bank..
Display segment always high or missing in displays of bank B.	See above; replace U20.

Digit missing or showing overlapping segments with digits in bank (displaying all segments that other digits have showing)	Probe appropriate inverter of U21, U22, or U23 for proper operation to digit. If incorrect replace chip, if correct replace U19.
No solenoids will fire	Check output pulses at pins 2, 5, 6, 9, 12, 15, 16, and 19 of U14 while in coil self-test. If no output is seen, check pin 11, of U14 for enable pulse. If no pulse is seen replace U8, else replace U14
Solenoid won't fire.	Check output pulse for solenoid at inverter outputs of U12 or U13 while in coil self-test, if pulse is absent, check input to corresponding driver gate to confirm proper operation from U14. If incorrect replace U14.
Solenoid always energized	Disconnect J5, if solenoid is still energized, driver board is bad. Otherwise probe U12, U13, and U14 as above.
CPU crashes/sporadically resets	Generally, this is NOT a CPU problem, but rather an associated problem with the +5 voltage level. It is only now being seen with the replacement CPU, because the old CPU used CMOS logic (12v) that was much more immune to voltage spikes/drop-outs. The problem can normally be reproduced by causing the CPU to energize a solenoid, while simultaneously energizing a non-CPU controlled solenoid (such as a bumper). This will stress the power supply enough to drop-out the voltage regulator or surge the ground. Check that all solenoids are the correct part number and that the snubbing diodes across them are working. Check for good ground connections at the power supply header pins. Check that the solenoid drive transistors are not "leaking". Check the connector between the CPU and driver board for good connection. Check the power supply voltage regulator for a solid +5 volts.
CPU looses memory settings inconsistently; game crashes occasionally.	Enter diagnostics, and reset the memory settings using the "InIT E" entry described above. Remove the solenoid fuse, and power down the game. If, after some time, the CPU correctly displays the default HSTD after power is re-applied, it is probable that the CPU is being crashed by a ground surge associated with a bad solenoid transisitor or missing snubbing diode across one of the playfield solenoids, as above. Check the knocker components carefully.
CPU consistently fails to power-up; depressing reset button does reset it correctly.	Check C11 for 47 micro-Farad capacitance, replace C11.

Warranty

The ***Ni-Wumpf*** replacement System 1 CPU is warranted against manufacturing defects and premature component failure, known as infant mortality, for up to 90 days after the date of it's purchase. For this reason, it may be prudent to keep a copy of the receipt for the board handy, in case the need should arrive to lend some credibility to your claims as to when it was actually purchased. In the unlikely event that a part should fail on the board within this time frame, simply package up the board, with a copy of your receipt, and return it to the factory address mentioned below, with some note of explanation as to why we are receiving this unexpected gift. Your CPU will be repaired or replaced and returned to you as quickly as possible.

Failure of the system within the warranty period due to abuse or incorrect installation, is not covered by this warranty. The purpose of this warranty should be quite clear: We are willing to accept responsibility for manufacturing defects of the boards we are building, and feel that 90 days is a sufficient period to determine if such a defect may exist. We are not willing to accept the responsibility of others who may install this board into a game that may be broken enough to cause the CPU to fail.

- Repair

To determine the steps required to repair your CPU, it is best to gather the information necessary to get the board serviced, whether by yourself or the factory. First of all, be certain that the CPU is actually broken before calling it in for service, and the best way to ascertain this is to consult this manual in detail under the configuration and troubleshooting sections. Next, armed with the confidence that a smoking chip is indeed reason for repair, contact your distributor to determine what steps to take next. If you intend to do the repair yourself, he will be the person with the parts necessary to replace the chips you deem necessary. He will also be the person with the knowledge of what repair costs the factory will be charging if you wind up sending it in for repair (and in the case that the company address should change, he will know what the current address for ***Ni-Wumpf*** will be).

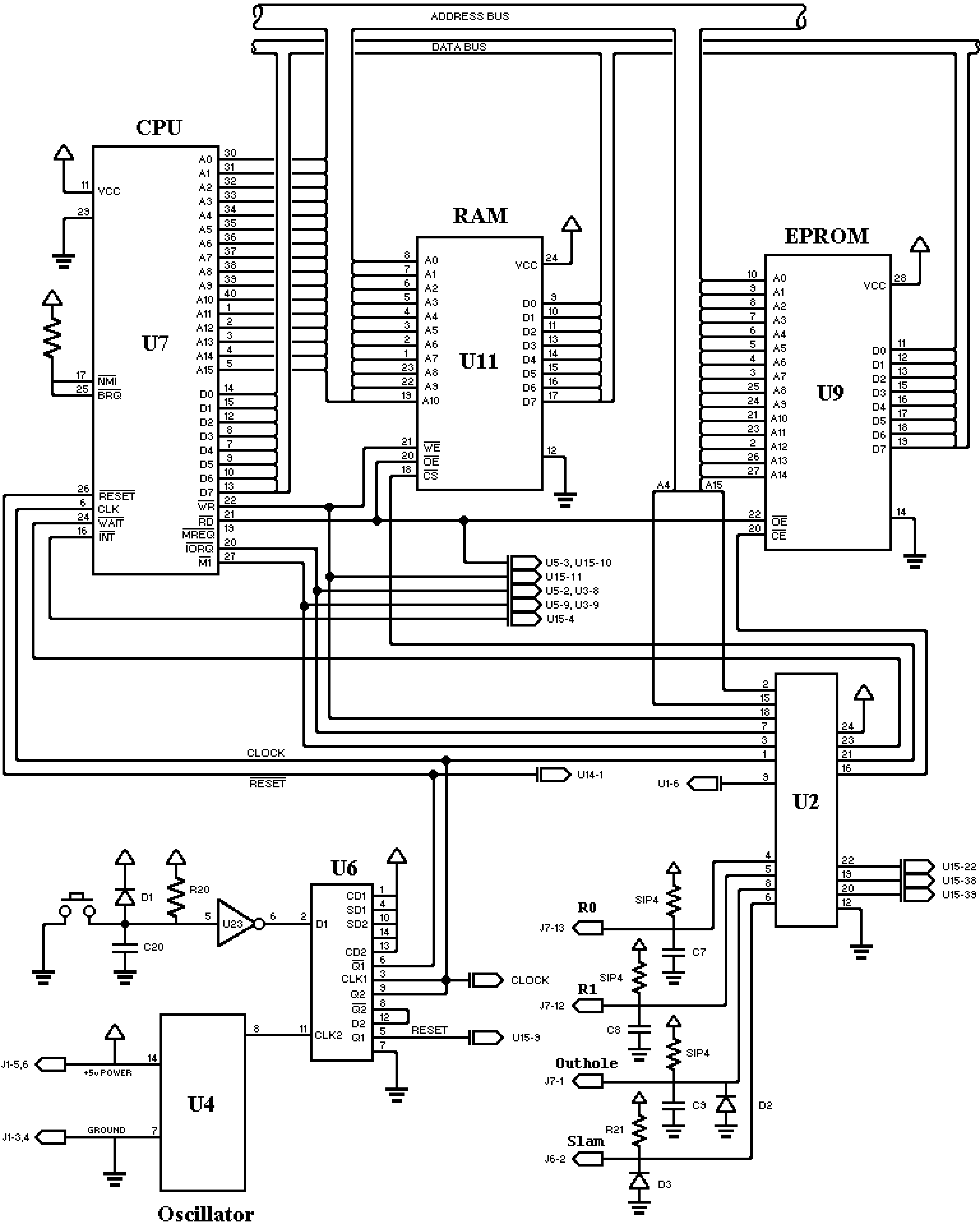
Repair of the ***Ni-Wumpf*** CPU is performed at the factory also, though there are a few options available for repair that may be exercised. Because the board design is relatively simple, it should be reasonable to undertake repair of the board yourself should you feel capable of doing so. There are only five PAL's, and one EPROM, that require factory programming (and one EEPROM that might be difficult to find at electronics stores), and these parts may all be purchased directly from the factory or your distributor if they are determined to be at fault. All of the rest of the parts are easily obtainable from different component sources. Now in the case of board repair being sent to the factory, return the board to our address below with a note describing the problem. Repair costs should be paid in advance and may be determined by contacting the factory beforehand.

Ni-Wumpf ltd.
4707 Clover St
Honeoye Falls, NY 14472-9306
(716) 582-2507

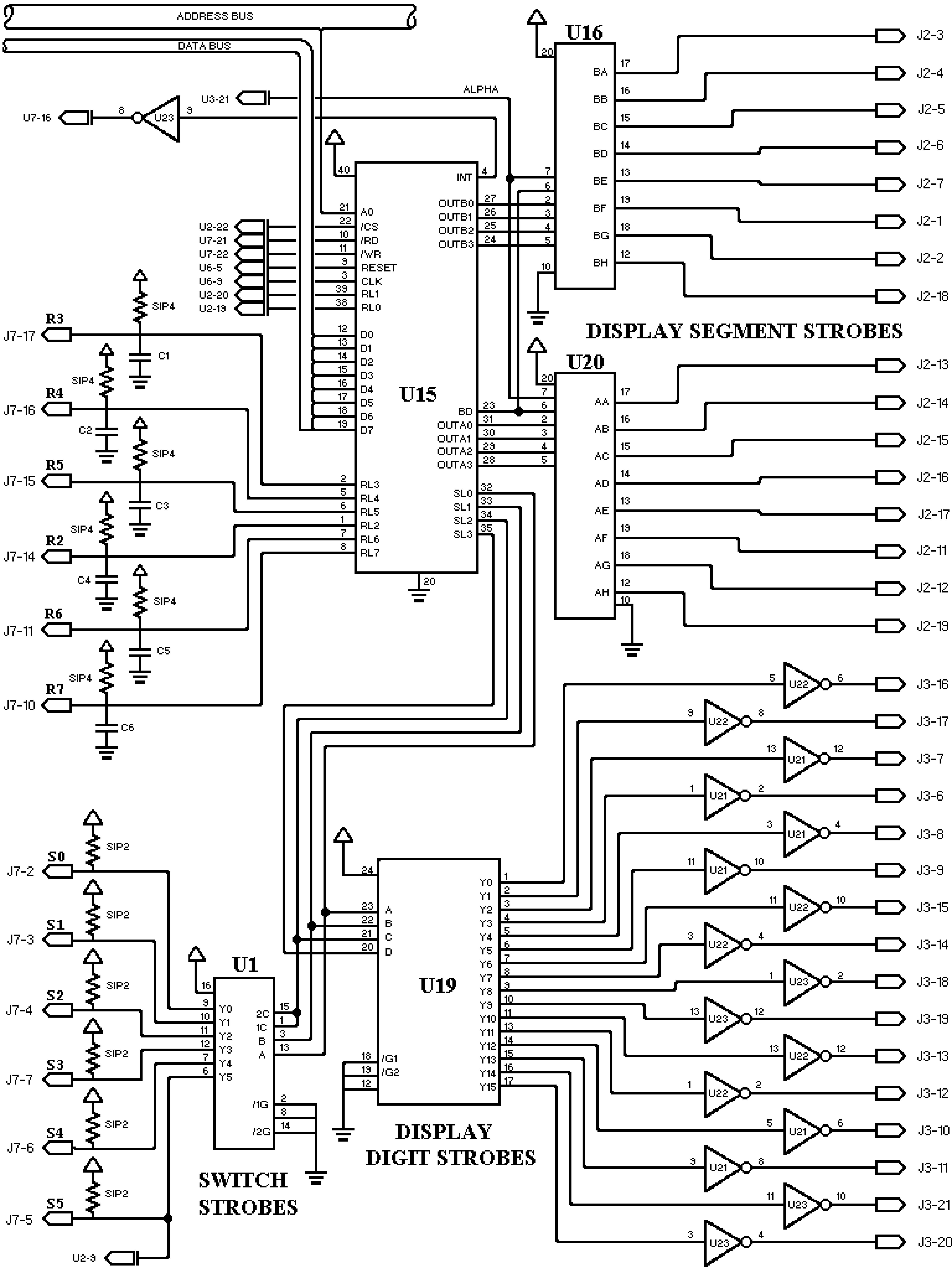
Schematic supplement for the
Ni-Wumpf System 1
replacement CPU

Part Designation	**Value**	**Description**
C0	100µf	Power filter cap.
C1 - C9	.01µf	Switch settling cap.
C16	47µf	Power filter cap.
C20	47µf	Reset cap.
D1 - D3	1N4149	Reset diode and bypass diodes
D4 - D15	1N5123	5.1v Zener diode
LED		LED
R1 - R8 or SIP 3	620Ω	Pull-up resistors
SIP 5	3.3kΩ	Pull-up lamp resistors[18]
R19	1kΩ	LED resistor[18]
R20	3.3kΩ	Reset circuit resistor[18]
R21	3.3kΩ	Outhole return resistor[18]
R22[20]	1kΩ	Pull-up resistor[18]
SIP1	2.2kΩ	Switch pull-up resistor[18]
SIP2, SIP4	3.3kΩ	Strobe/Returns pull-up resistors[18]
U1	74156	3 to 8 line demultiplexer
U2, U3, U5N	22V10	PAL
U4	3.276Mhz	Oscillator
U5, U16, U20	16L8	PAL
U6	74HCT74	Flip-flop
U7	Z80	CPU
U8[22]	18V8	PAL
U9	27256	Game EPROM
U10	93C06/C46	EEPROM
U11	6116	2k RAM
U12, U13	7416	Inverter drivers
U14	74374	Solenoid latch
U15	8279	PIA
U17	74244	Bus driver
U18		Dip Switch
U19	74154	4 to 16 line demultiplexer
U21, U22	7404	Lamp strobe inverter drivers
U23	74LS14	Lamp strobe inverter driver
SW1		Reset switch
-	bypass cap.'s	All bypass capacitors are .1µf

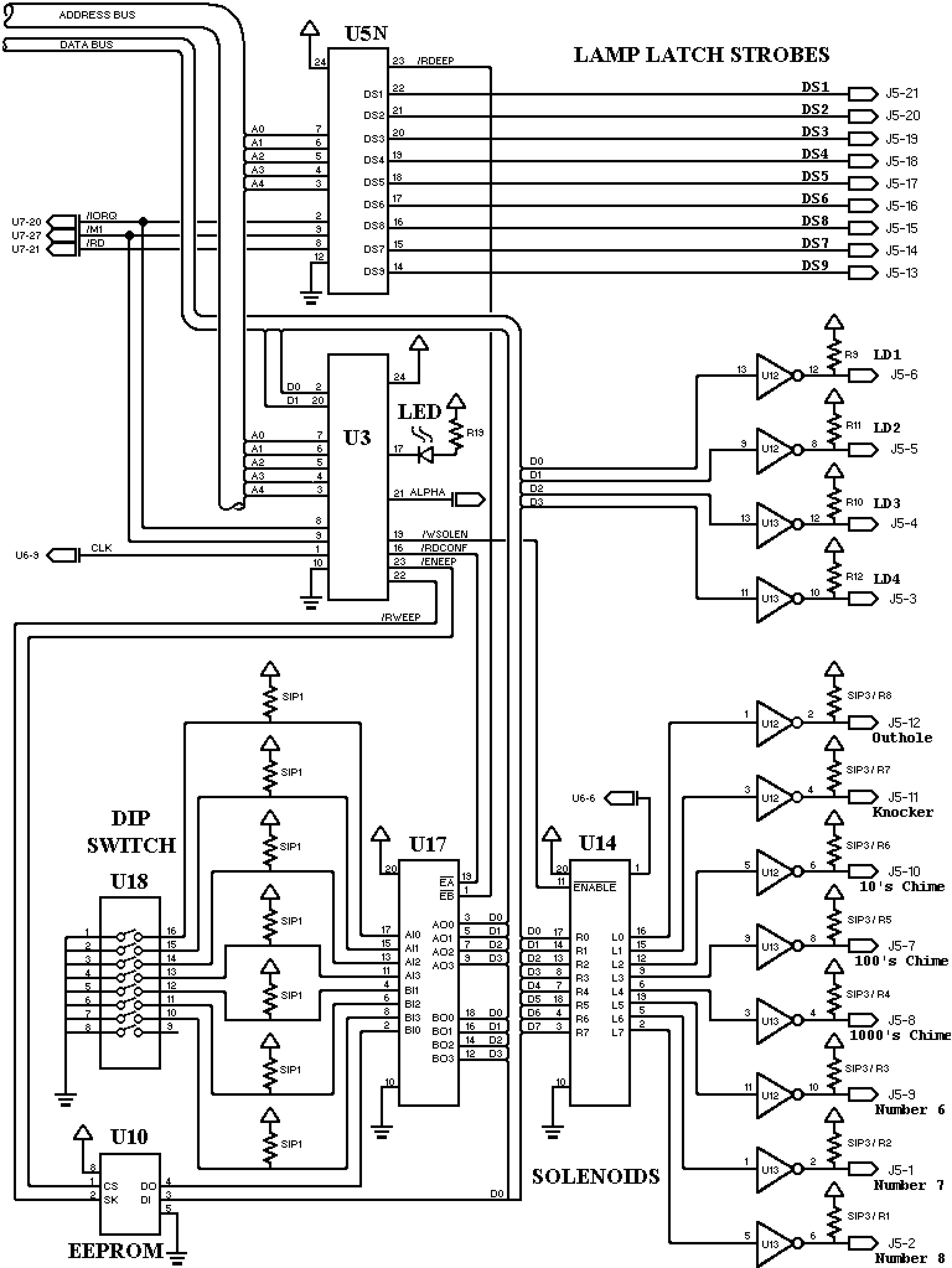
Table 6 - Table of parts on the Ni-Wumpf CPU



ADDRESS BUS
DATA BUS
CPU
U7
RAM
U11
EPROM
U9
U2
U6
U4
Oscillator
CLOCK
RESET
U14-1
U1-6
U5-3, U15-10
U15-11
U5-2, U3-8
U5-9, U3-9
U15-4
U15-22
U15-38
U15-39
U15-3
R0
R1
Outhole
Slam
J7-13
J7-12
J7-1
J6-2
SIP4
R20
R21
D1
D2
D3
C7
C8
C9
C20
U23
J1-5,6
J1-3,4
+5v POWER
GROUND



ADDRESS BUS
DATA BUS
ALPHA
U3-21
U7-16
U23
U16
BA BB BC BD BE BF BG BH
J2-3 J2-4 J2-5 J2-6 J2-7 J2-1 J2-2 J2-18
DISPLAY SEGMENT STROBES
U15
INT
A0 /CS /RD /WR RESET CLK RL1 RL0
U2-22 U7-21 U7-22 U6-5 U6-9 U2-20 U2-19
D0 D1 D2 D3 D4 D5 D6 D7
OUTB0 OUTB1 OUTB2 OUTB3
BD OUTA0 OUTA1 OUTA2 OUTA3
SL0 SL1 SL2 SL3
RL3 RL4 RL5 RL2 RL6 RL7
R3 R4 R5 R2 R6 R7
J7-17 J7-16 J7-15 J7-14 J7-11 J7-10
SIP4
C1 C2 C3 C4 C5 C6
U20
AA AB AC AD AE AF AG AH
J2-13 J2-14 J2-15 J2-16 J2-17 J2-11 J2-12 J2-19
U21 U22 U23
J3-16 J3-17 J3-7 J3-6 J3-8 J3-9 J3-15 J3-14 J3-18 J3-19 J3-13 J3-12 J3-10 J3-11 J3-21 J3-20
U19
A B C D
Y0 Y1 Y2 Y3 Y4 Y5 Y6 Y7 Y8 Y9 Y10 Y11 Y12 Y13 Y14 Y15
/G1 /G2
DISPLAY
DIGIT STROBES
U1
Y0 Y1 Y2 Y3 Y4 Y5
2C 1C B A
/1G /2G
SWITCH
STROBES
S0 S1 S2 S3 S4 S5
J7-2 J7-3 J7-4 J7-7 J7-6 J7-5
SIP2
U2-9


ADDRESS BUS
DATA BUS
U5N
LAMP LATCH STROBES
/RDEEP
DS1 J5-21
DS2 J5-20
DS3 J5-19
DS4 J5-18
DS5 J5-17
DS6 J5-16
DS8 J5-15
DS7 J5-14
DS9 J5-13
U7-20 /IORQ
U7-27 /M1
U7-21 /RD
U3
LED
R19
ALPHA
/WSOLEN
/RDCONF
/ENEEP
/RWEEP
U6-9 CLK
LD1 R9 U12 J5-6
LD2 R11 U12 J5-5
LD3 R10 U13 J5-4
LD4 R12 U13 J5-3
SIP3 / R8 U12 J5-12 Outhole
SIP3 / R7 U12 J5-11 Knocker
SIP3 / R6 U12 J5-10 10's Chime
SIP3 / R5 U13 J5-7 100's Chime
SIP3 / R4 U13 J5-8 1000's Chime
SIP3 / R3 U12 J5-9 Number 6
SIP3 / R2 U13 J5-1 Number 7
SIP3 / R1 U13 J5-2 Number 8
DIP SWITCH
U18
SIP1
U17
U14
U6-6
ENABLE
U10
CS SK DO DI
EEPROM
SOLENOIDS