A newsletter about machine control, automation, business and the Universe in general. Dedicated to the proposition that SPLat happens.

SPLat Happenings

April 2000

Free to all registered SPLat customers world-wide.
Also available free on our website and by email.

SPLat gets even easier

Fast-track makes it happen

A new programming concept for SPLat, called Fast-track, has made SPLat even easier than ever to program. Fast-track is arguably the easiest method of control programming ever devised. It can reduce the time a novice requires to make useful programs from *several* hours to a *couple* of hours.

The Fast-track idea came to us as a result of observing the ways in which our customers were using SPLat for real-world tasks. What we observed was that SPLat was in many cases being "under-utilised". Essentially, most users are using SPLat for what we call "linear" control sequences. A linear sequence is one where the control performs a fixed, set sequence of steps, with each step triggered by an external event (say a limit switch or manual push button) or by an internal time-out. This contrasts with what we would have to call a non-linear sequence, which contains many alternative actions and re-actions depending on external circumstances.

Further observation of these linear programs revealed that they all contained frequent occurrences of the same code sequences, namely

Continued over ...

...the easiest method of control programming ever devised.

SPLat training happens!

The first formal training classes for SPLat users were run late February and early March. It is our intention to develop a comprehensive system of training, both as an in-house initiative and in concert with major tertiary education institutes (expect a significant announcement about the latter in the next SPLat Happenings!)

The first class was run at the Melbourne offices of AVH, our Australian distributor. It was a "beginners" class, starting with "what is a controller?" and moving up from there. In two days we got the students up to the stage of understanding not only SPLat but some of the golden "Universal Truths" like the importance of good specification and structured program design. We firmly believe that when we are wearing our educators' hats we should provide depth of knowledge, not just SPLat-specific information.

The following week we ran an intensive one-day class in Shepparton, a town in regional Victoria. About 10 contractors and plant engineers came from up to 3 hours away. This was billed as an "intermediate" level class, assuming the students had pre-studied Fast-track and at least scanned the tutorial. Again, "Universal Truths" were a large part of the day's work.

Making it happen in Shepparton

SPLat consultants wanted!

Do you believe you have the skills and talents necessary to offer your services as a SPLat applications consultant? As SPLat sales continue to grow we cannot possibly design solutions and write programs for every new customer. Hence, we are building a register of suitable people to whom we can refer customers who lack the skills or the time to do their own programming and electrical design. Needs vary all the way from simply writing a program to supplying a turn-key solution. There are even times when a customer needs an ongoing supply of fully built up control systems based on SPLat, month in, month out. The business opportunities are attractive for the right kind of person or company, so if this interests you please drop us a line to discuss it.

Your Business

www: key to massive sales?

SPLat have had a web site for over 4 years. In Internet terms that's a longtime. So, what is our experience of having a web presence?

If you're not on the web, you will still be in business in 5 years. The Internet holds the promise of exiting new ways of doing business, but for many it is just that: a promise. The reality is that people will still buy fruit from fruit shops and clothes from clothes shops, and they will still look for an electrician in the local paper or yellow pages.

Before you set up a web site ("home page"), work out exactly what you want it to achieve. Is your web site there to make sales. Is it there as a corporate statement? (Durex have a *huge* web site, but they don't take on-line orders for their condoms.) Or is your web site there to provide technical support for your customers, say technical tips, software updates or a customer to customer chat room?

Web package deals are a waste of money. A web page package deal is about like having a letter sized poster glued to the outside wall of the top story of a skyscraper. No-one will see it except perhaps a passing hot air balloonist. Sure, it's theoretically visible to the whole population of the city, but will anyone notice it? With most package deals the promoter will suggest that you can simply pay over some hard-earned cash and give them some words and a picture or two. Your web site will then be created for you and you can just sit back and rake in the orders. Just ask yourself this: Is the promoter out to sell his product or yours? In reality a web site is a dynamic, evolving medium, and any web site plan must include a plan for how you will achieve constant revisions and updates. Either you learn how to do it yourself, or you engage someone to create and maintain it for you.

A web site is just one part of the marketing mix. You should have an overall marketing plan before you rush out and mount a web site.

If you are interested in web marketing, do your homework first. Start by subscribing to several of the excellent free email newsletters. A good place to start is www.wilsonweb.com, which contains a massive amount of web marketing information.

Got your ABN yet?

Did you know that all businesses in Australia must have their A.B.N. in place by July 1? If you trade as a business and do not have an A.B.N your business clients will be obliged to withhold 48.5% tax from your payments. People will be forced out of business because of it. Contact the tax office now! Applications close May 31.

Product codes rationalized

Over the last few months we have come to realize that the product designation on the various SPLat boards were becoming a little inconsistent. Clearly you can't always get it right the first time! To try and make things a bit simpler for our distributors, resellers and end-users we have therefore changed the model numbers on most products in line with a consistent scheme.

Henceforth all controllers and add-on boards will be offered in two versions, with different suffices. These are:

-DK This is the Developer's Kit version, and is the form in which a board is always sold as a one-off. The DK will contain the board plus other items (varying from product to product) needed to get going on a development project.

-OEM This is the board sold on its own with no extras, and intended for quantity OEM use. The OEM versions have a Minimum Order Quantity (MOQ) of 5 units.

We have also re-badge the SP10-8ANC to SP108 and made the DIN mount base an option, and removed the -A suffix from the SL88.

The Fast-track list

There are just 11 Fast-track instructions. These, plus 3 GoTo instructions form a set of 14 instructions that are all you need to write a range of useful controller programs with SPLat. The 14 instructions are :

WaitON Wait for a nominated input to turn on.

WaitOff Wait for a nominated input to turn off.

Pause Pause program execution for a nominated period (10mS to 327 seconds)

GoIfInOn Go to a specified program line if a nominated input is on.

GoIfInOff Go to a specified program line if a nominated input is off.

WaitOnT Wait for a nominated input to turn on, but with a time limit. (10mS to 327 seconds) The following instruction can test the outcome (the input happened or the instruction timed out) and do a conditional GoTo based on the result.

WaitOffT Wait for a nominated input to turn off, but with a time limit. Testing as above.

ON Turn on a nominated output

OFF Turn off a nominated output

SetMem Set a memory to a preset count

DecMGoIfNZ Decrement (count down) a memory (counter), do a goto if it's non-zero (i.e. not yet counted down to zero)

GoTo Unconditionally go to a specified (named) program line.

GoIfT Go to a specified (named) program line if the previous operation gave a True result (typically if a WaitOnT got the expected input)

GoIfF Go to a specified (named) program line if the previous operation gave a False result (typically if a WaitOnT did not get the expected input and instead timed out).

Programming

Getting results with Fast-track

Fast-track is the new SPLat programming concept that makes it incredibly easy to program linear control sequences. This article shows how easy it can be to get results using Fast-track.

The best way to illustrate the use of Fast-track is with an example. The example is a controller for a spa bath pump. It has two push buttons and a pump output. If button A is pressed the pump comes on. If button B is pressed it turns off. If both buttons are pressed it pulsates 1 second on, 1 second off. Here's the program (the instructions are defined in the box "The Fast-track list"):

```
Top       Off       Pump
          WaitOff   A
          WaitOff   B
Idle      WaitOn    A
          Pause     25
          GoIfInOn  B,Pulse

JustOn    On        Pump
          WaitOn    B
          Off       Pump
          GoTo      Top

Pulse     On        Pump
          WaitOff   B
PulsAgin  Off       Pump
          WaitOnT   B,100
          GoIfT     Top
```

At the Top of the program we make sure the pump and both buttons are off. We then wait for button A to be pressed at the line labeled Idle. Once we have the A button we give the user 1/4 second (Pause 25) to press button B to achieve pulse mode. If button B is pressed after that time we GoTo the Pulse function, otherwise we simply turn on at JustOn. At JustOn we turn on the pump then wait for button B before turning off the pump and starting the whole process gain by going back to Top.

Meanwhile, to initiate pulse mode we find ourselves at Pulse. Here we turn on the pump and then wait for button B to go off (this is because the next time B comes on it is to stop the pump totally). The pulse function is a loop starting at PulsAgin and ending at the bottom of the program. It uses WaitOnT instructions as timers that can be interrupted by button B. In one instance it times the on period and in the other the off period of the pump. If button B is pressed the GoIfT will transfer control back to Top.

Fast-track makes it happen

From page 1

small program loops that wait for something to happen, be it an input or a time-out.

The other factor that led to Fast-track was that we have been conducting a lot of trade nights lately, where we present and explain SPLat to 10-20 electrical contractors and other potential SPLat users. In these trade nights we show a bit of an example of programming. This means we have 10-15 minutes to explain how the programming works and to get our listeners to grasp the basic principles. Very often our listeners have absolutely no prior exposure to programming. This was where the repeated program loops became painfully obvious as something that was necessary for the mechanics but not for the logic of the program. For example, it's OK to explain that "we wait for the input to come on", but why have to explain that this means repeatedly reading the input, testing if it's on and going back to read it again if it's not on?

... make useful programs using only 5 instructions...

Enter Fast-track. What we did was to identify all the commonly used program loops and then create a single instruction to replace each one. That way one Fast-track instruction replaces up to 6 regular instructions and at the same time makes SPLat programming substantially easier to understand (and for us, easier to teach).

We set another criterion for Fast-track: It had to be possible to make useful applications of SPLat without needing any knowledge of the internal workings of SPLat. Under this definition we re-badged the old ON and OFF instructions as Fast-track.

There are a total of 11 Fast-track instructions. These, combined with 3 GoTo instructions form a basic repertoire of just 42 instructions that are all that are needed to make useful programs with SPLat. You can actually make useful programs using only 5 instructions. See "Getting results with Fast-track" for an example of a Fast-track program.

We also added a new Fast-track section at the start of the tutorial. It has 10 chapters (screens) and is designed to get you up and running in just a couple of hours. All you need to know is how to drive a Windows computer and the electrical skills to wire a SPLat controller into your circuit.

A rose by any other name...

In SPLat/PC V4.7 we introduced not only Fast-track programming, but also some instruction aliases. An alias is an alternative name for an instruction that is intended to make it easier to understand.

Specifically we had come to realise that GoIfNZ and GoIfZ were at times a little obscure. Hence from now on you can use GoIfT (go if true) or GoIfF (go if false) and get exactly the same results.

Oops!

In the last issue we accidentally mis-quoted the price for the RLX8 and IOX44 add-on boards. The price quoted was for the -OEM version. -DK versions are $A149, including a 10-way to 10-way cable and instructions

Published by SPLat Controls Pty Ltd
2/12 Peninsula Blvd Seaford VIC 3198, AUSTRALIA
www.splatco.com.au **happenings@splatco.com.au**
Ph +61 3 9773 5082